Bellabeat Project

Imber C

2024-02-17

Bellabeat Capstone Project

Project Premise:

As part of Google Data Analytics course, I am doing a case study in the role of a junior analyst for Bellabeat, a company that manufactures health-focused products for women.

Urška Sršen, cofounder and Chief Creative Officer of Bellabeat wants us to analyze smart product information from other brands so that we can discover growth opportunities. They want us to focus on one Bellabeat product and gain insight from the data we are analyzing to suggest and inform marketing strategies. Sršen wants us to analyze smart device user data and apply these insites to a Bellabeat product.

These are the questions that will guide our 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?

This is the information I have been given about the products Bellabeat currently offers:

-————————————————————————————-

Bellabeat app: The Bellabeat app provides users with health data related to their activity, sleep, stress, menstrual cycle, and mindfulness habits. This data can help users better understand their current habits and make healthy decisions.The Bellabeat app connects to their line of smart wellness products.

Leaf: Bellabeat's classic wellness tracker can be worn as a bracelet, necklace, or clip. The Leaf tracker connects to the Bellabeat app to track activity, sleep, and stress.

Time: This wellness watch combines the timeless look of a classic timepiece with smart technology to track user activity, sleep, and stress. The Time watch connects to the Bellabeat app to provide you with insights into your daily wellness.

Spring: This is a water bottle that tracks daily water intake using smart technology to ensure that you are appropriately hydrated throughout the day. The Spring bottle connects to the Bellabeat app to track your hydration levels.

Bellabeat membership: Bellabeat also offers a subscription-based membership program for users. Membership gives users 24/7 access to fully personalized guidance on nutrition, activity, sleep, health and beauty, and mindfulness based on their lifestyle and goals." -————————————————————————————

Breaking this down, we know Bellabeat has the ability to track:

- **Activity**

- **Sleep**
- **Stress**
- **Menstrual Cycle**
- **Mindfulness Habits**
- **Hydration**

We have been given a data set from Fitbit so we can compare what other apps/producst on the market are tracking:

A variety of tables have been provided, including Heartrate per second,Calories, Intensities, Steps, and Sleep recorded at the minute level, Calories, intensities, steps, combined to the hourly level, and Activity, Calories, Steps, Sleep, and Intensities recorded at the day level, and a table of weights entered for some users throughout the time period.

The data taken at the second or minute level will be too detailed for our analysis, but it is useful to know that this is the level of detail other apps are using, to be sure Bellabeat matches that level with our user interface.

These datasets include

- **Heartrate**
- **Calories**
- **Activity with intensities**
- **Steps**
- **Sleep**
- **Weight**

To start, we prepare our packages and our location where we have our data: I installed the following packages: tidyverse, here, skimr, janitor, ggplot2, and dplyr.

Next I bring in all the tables and take a look at exactly what data is provided for us to work with.

```
head(Dactivity)
```

```
## # A tibble: 6 x 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
##        <dbl> <chr>             <dbl>         <dbl>           <dbl>
## 1 1503960366 4/12/2016         13162          8.5             8.5
## 2 1503960366 4/13/2016         10735          6.97            6.97
## 3 1503960366 4/14/2016         10460          6.74            6.74
## 4 1503960366 4/15/2016          9762          6.28            6.28
## 5 1503960366 4/16/2016         12669          8.16            8.16
## 6 1503960366 4/17/2016          9705          6.48            6.48
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

```
head(Dintensities)
```

```
## # A tibble: 6 x 10
##         Id ActivityDay SedentaryMinutes LightlyActiveMinutes FairlyActiveMinutes
##      <dbl> <chr>                  <dbl>                <dbl>                <dbl>
```

```
## 1   1.50e9 4/12/2016               728                  328                  13
## 2   1.50e9 4/13/2016               776                  217                  19
## 3   1.50e9 4/14/2016              1218                  181                  11
## 4   1.50e9 4/15/2016               726                  209                  34
## 5   1.50e9 4/16/2016               773                  221                  10
## 6   1.50e9 4/17/2016               539                  164                  20
## # i 5 more variables: VeryActiveMinutes <dbl>, SedentaryActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, moderatelyActiveDistance <dbl>,
## #   VeryActiveDistance <dbl>
```

```
head(Dsteps)
```

```
## # A tibble: 6 x 3
##           Id ActivityDay StepTotal
##        <dbl> <chr>           <dbl>
## 1 1503960366 4/12/2016       13162
## 2 1503960366 4/13/2016       10735
## 3 1503960366 4/14/2016       10460
## 4 1503960366 4/15/2016        9762
## 5 1503960366 4/16/2016       12669
## 6 1503960366 4/17/2016        9705
```

```
head(Dcalories)
```

```
## # A tibble: 6 x 3
##           Id ActivityDay Calories
##        <dbl> <chr>          <dbl>
## 1 1503960366 4/12/2016       1985
## 2 1503960366 4/13/2016       1797
## 3 1503960366 4/14/2016       1776
## 4 1503960366 4/15/2016       1745
## 5 1503960366 4/16/2016       1863
## 6 1503960366 4/17/2016       1728
```

```
head(weight)
```

```
## # A tibble: 6 x 8
##           Id Date       WeightKg WeightPounds   Fat   BMI IsManualReport   LogId
##        <dbl> <chr>         <dbl>        <dbl> <dbl> <dbl> <lgl>            <dbl>
## 1 1503960366 5/2/2016 ~     52.6         116.    22  22.6 TRUE           1.46e12
## 2 1503960366 5/3/2016 ~     52.6         116.    NA  22.6 TRUE           1.46e12
## 3 1927972279 4/13/2016~    134.          294.    NA  47.5 FALSE          1.46e12
## 4 2873212765 4/21/2016~     56.7         125.    NA  21.5 TRUE           1.46e12
## 5 2873212765 5/12/2016~     57.3         126.    NA  21.7 TRUE           1.46e12
## 6 4319703577 4/17/2016~     72.4         160.    25  27.5 TRUE           1.46e12
```

```
head(sleep)
```

```
## # A tibble: 6 x 5
##           Id SleepDay        TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##        <dbl> <chr>                       <dbl>              <dbl>          <dbl>
```

```
## 1 1503960366 4/12/2016 12:0~                1                327                346
## 2 1503960366 4/13/2016 12:0~                2                384                407
## 3 1503960366 4/15/2016 12:0~                1                412                442
## 4 1503960366 4/16/2016 12:0~                2                340                367
## 5 1503960366 4/17/2016 12:0~                1                700                712
## 6 1503960366 4/19/2016 12:0~                1                304                320
```

Right away we can notice that data is seperated by user Id and date. The date columns are different formats and have different names. It would be a good idea to streamline date format and column names at this time. However, I did not do this upfront.

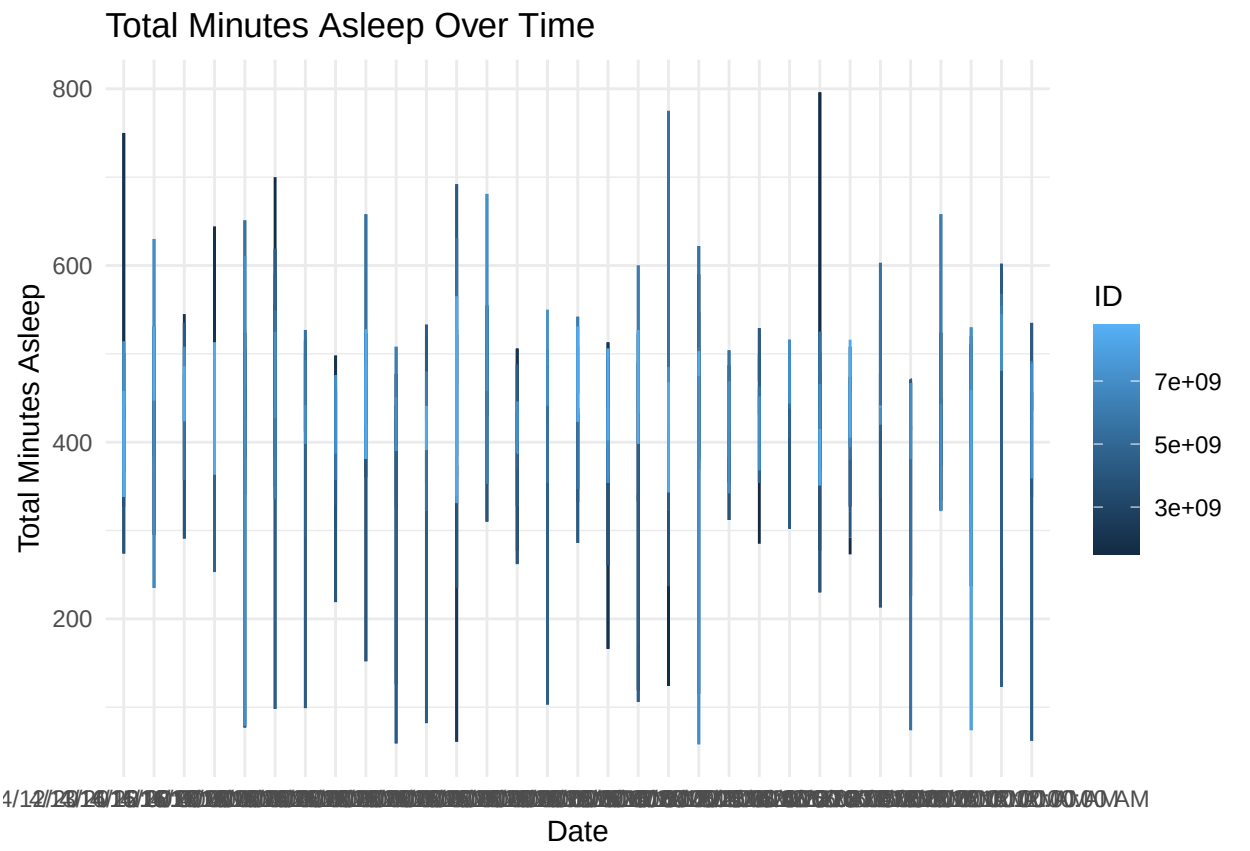
Other interesting information:

- Sleep measures time in bed and time asleep.
- Weight shows fat% and BMI, and includes options to manually report- which means some measurements are not manually reported. This could be a huge area of potential growth for Bellabeat to add a scale to it's product line. Currently weight is not tracked by Bellabeat products at all.
- Dsteps is a simple dataset of Id, ActivityDay, and StepTotal. Bellabeat doesn't currently track steps at all, which is definitely an oversight. It is one of the simplest activities to track for even the most casual users. The Steps dataset is contained within the Dactivities dataset but may be easier to use alone.
- Calories is a simple table of ID, ActivityDay, and Calories. I thought this might be Calories in, since Calories also appears in the Dactivities but after laboriously combining the two dataframes, I discovered they were identical numbers. I should have sorted the data and taken a better look first- lesson learned!
- Daily activity and intensities have four different levels of activity. We want to be sure we are able to break down user activity into at least as many detailed brackets.
- Hourly intensities may prove useful to Bellabeat in encouraging users to work out during their normal active times.

Sleep and Mindfulness

I want to start with the sleep dataset. I would like to graph the difference between time in bed and time asleep. The idea here is using Bellabeat mindfullness routines, users should be able to see a difference in the amount of time spent in bed, but not asleep. Let me start by seeing how complicated the graph would be if we tried to see all users:

```
ggplot(sleep, aes(x = SleepDay, y = TotalMinutesAsleep, color = Id)) +
  geom_line() +
  labs(title = "Total Minutes Asleep Over Time",
       x = "Date",
       y = "Total Minutes Asleep",
       color = "ID") +
  theme_minimal()
```



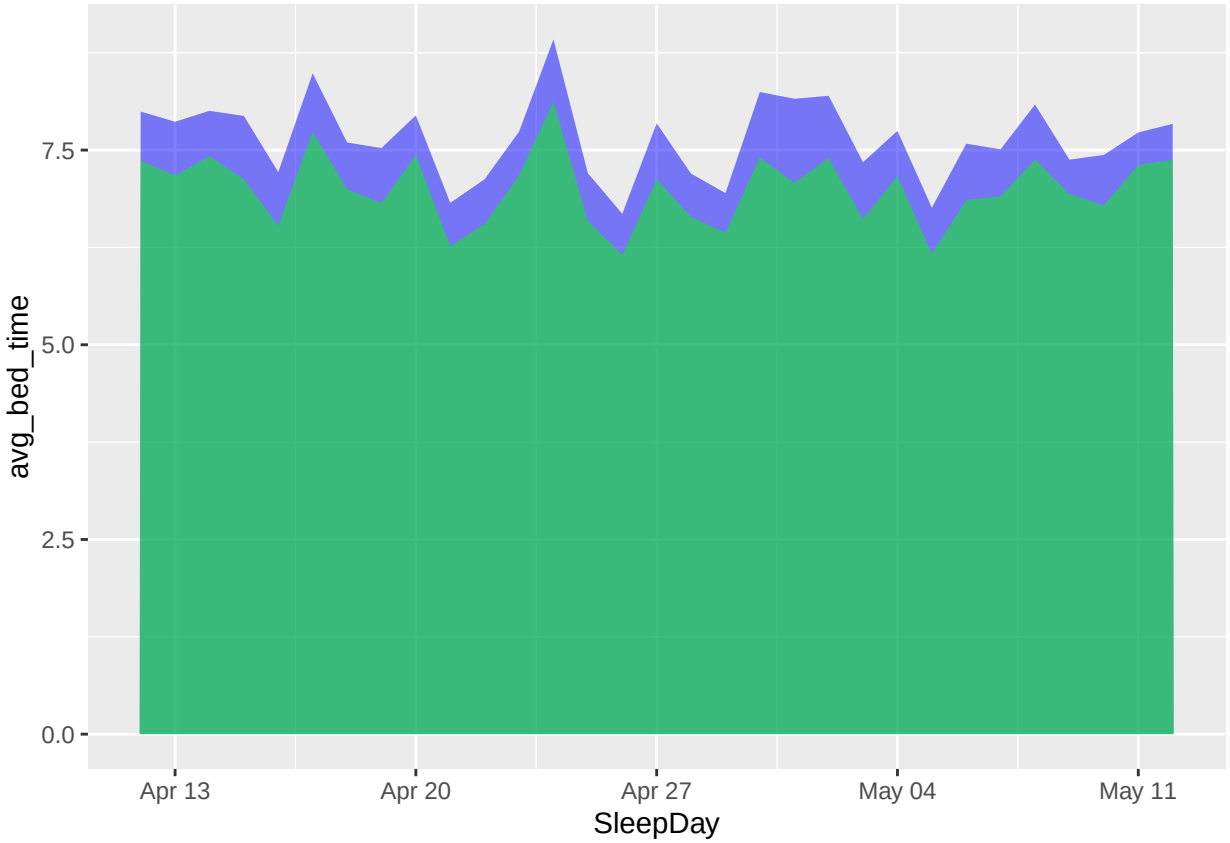
Right away, this is overly complicated. An average over all Id's would be a simplified example. I also think minutes asleep should be converted to hours for ease of understanding.

So I want first to convert minutes into hours, then make sure my date is formated as a date, then calculate average sleep time for each SleepDay.

```
sleep <- sleep %>%
  mutate(SleepHours = TotalMinutesAsleep / 60) %>%
  mutate(BedHours=TotalTimeInBed/60)

sleep$SleepDay<-as.Date(sleep$SleepDay, format ="%m/%d/%y")

average_sleep <-sleep %>%
  mutate(SleepHours= (TotalMinutesAsleep / 60),
         BedHours = (TotalTimeInBed / 60)) %>%
  group_by(SleepDay) %>%
  summarise(avg_sleep_time=round(mean(SleepHours),digits=3),
            avg_bed_time=round(mean(BedHours), digits=3))
```

Now we should be ready to make a visualization. We are getting an average of all users, but we can also expect it is a good approximation of a single users results:

```
ggplot(average_sleep, aes(x=SleepDay))+
  geom_area (aes (y=avg_bed_time), fill="blue", alpha = 0.5)+
  geom_area (aes(y=avg_sleep_time),fill="green",alpha=0.5)
```



Suggested use for this data: This visual makes it very easy to see time spent in bed vs asleep. Since mindfullness is one of the things Bellabeat is trying to bring to our consumers, this will be nice to be able to track, hopefully showing restless time in bed decreasing after engaging with mindfullness routines.

Calories

Now let's look at calories:

Below is what I did to reformat and combine Calories from Dactivity and Dcalories, before I realized it was the same data:

str(Dcalories) ActivityDay is a chr string here as well.

Dcalories$ActivityDay<-as.Date(Dcalories$ActivityDay, format ="%m/%d/%y") Dactivity$ActivityDate<-as.Date(Dactivity$ActivityDate, format ="%m/%d/%y")

I want to join by Date and Id, and find it is easier to rename both fields that contain dates.

Dcalories<-rename(Dcalories, Date=ActivityDay) Dactivity<-rename(Dactivity, Date=ActivityDate) Dactivity<-rename(Dactivity, Calories_out=Calories)

Calories_in_out<-Dcalories %>% left_join(select(Dactivity, Date, Id, Calories_out), by=c("Date", "Id"))

Only after doing all of this did I realize the Calorie data was identical. I should have sorted my data before comparing, but the top lines were different and I assumed we had calories in and calories out.

Suggested use for this information: While I was not able to compare calories in to calories out, this does point out a hole in the data that other companies are collecting, Calories in is an important part of health and fitness.

Most active usage times

Next, I would like to see what hours users are most active, using Hintensities.

I first used this code to separate:

str(Hintensities) Hintensities3 <- Hintensities %>% separate(ActivityHour, into = c("date", "time"), sep= " ")

We lose AM/PM distinction with the above code, so I am going to change it to 24 hour time format.

```
Hintensities24 <- Hintensities %>%
  mutate(datetime = as.POSIXct(ActivityHour, format = "%m/%d/%Y %I:%M:%S %p"),
         datetime = ifelse(format(datetime, "%H:%M:%S") == "00:00:00",
                           "24:00:00", format(datetime, "%H:%M:%S")))
```

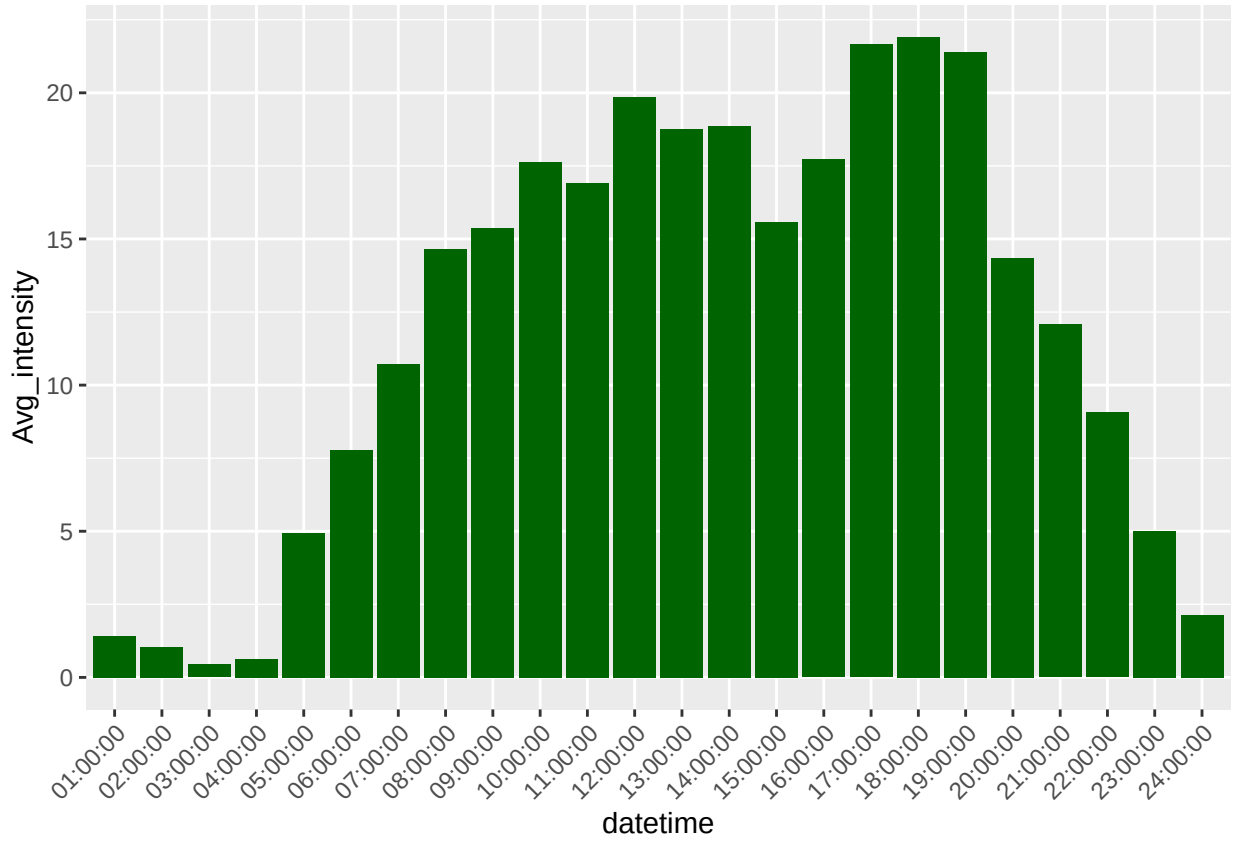
This adds a column with 24 hour time, which we can use to analyze the data.

We will group activity by datetime and summarize by intensity.

```
Hintensities2 <- Hintensities %>%
  mutate(datetime = as.POSIXct(ActivityHour, format = "%m/%d/%Y %I:%M:%S %p"),
         datetime = ifelse(format(datetime, "%H:%M:%S") == "00:00:00",
                           "24:00:00", format(datetime, "%H:%M:%S"))) %>%
  group_by(datetime) %>%
  summarise(Avg_intensity = mean(TotalIntensity))
```

Now we can graph the above to get a visual of peak useage times:

```
ggplot(Hintensities2, aes(x=datetime, y=Avg_intensity))+
  geom_col(fill="darkgreen")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Suggested use for this data: We can see there are a few peak usage times for the average user- Bellabeat can start sending nudges to users during those times, but can also adjust based on individual users usage once we have that information.

Activity intensity by time and distance

I would like to take a look at activity intensities with both distance and time, to see what conclusions we can draw:

```
Dactivity<-rename(Dactivity, Date=ActivityDate)
Dactivity$Date <- as.Date(Dactivity$Date, format = "%m/%d/%Y")
```

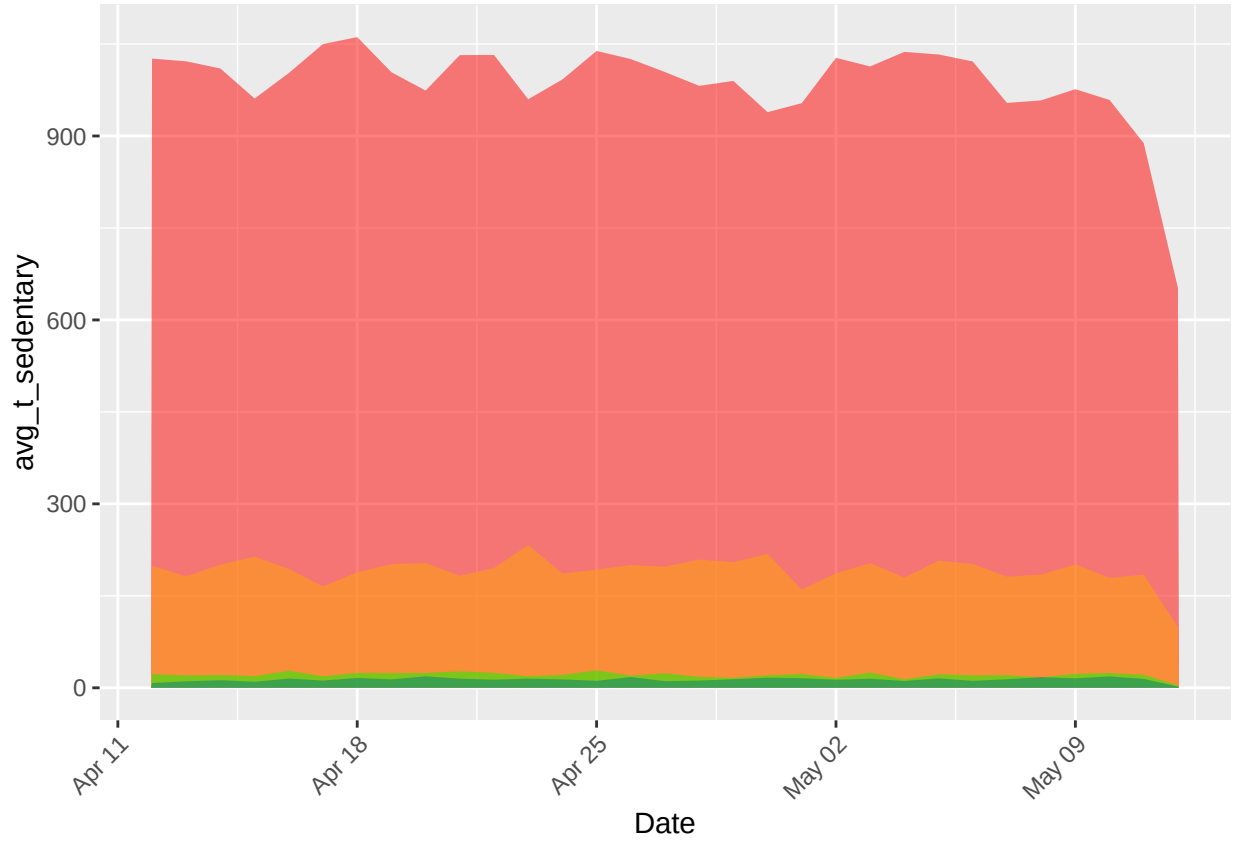
```
Intensity_distance<-Dactivity %>%
  group_by(Date) %>%
  summarise(avg_sedentary=round(mean(SedentaryActiveDistance), digits=2),
            avg_light=round(mean(LightActiveDistance),digits=2),
            avg_moderate=round(mean(ModeratelyActiveDistance),digits=2),
            avg_veryactive=round(mean(VeryActiveDistance),digits=2))

Intensity_time<-Dactivity %>%
  group_by(Date) %>%
  summarise(avg_t_sedentary=round(mean(SedentaryMinutes), digits=2),
            avg_t_light=round(mean(LightlyActiveMinutes),digits=2),
            avg_t_moderate=round(mean(FairlyActiveMinutes),digits=2),
            avg_t_veryactive=round(mean(VeryActiveMinutes),digits=2))
```

I was having some trouble graphing the above and realized my dates were not in proper format. A quick correction:

```
Intensity_distance$Date<-as.Date(Intensity_distance$Date, format="%m/%d/%y")
Intensity_time$Date<-as.Date(Intensity_time$Date, format="%m/%d/%y")
```

Activity time is still in minutes, we may consider changing to hours for some visuals later. Lets see how intensity looks over time:

```
ggplot(Intensity_time, aes(x=Date))+
  geom_area (aes (y=avg_t_sedentary), fill="red", alpha = 0.5)+
  geom_area (aes(y=avg_t_light),fill="orange",alpha=0.5)+
  geom_area (aes(y=avg_t_moderate),fill="blue",alpha=0.5)+
  geom_area (aes(y=avg_t_veryactive),fill="green",alpha=0.5)+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

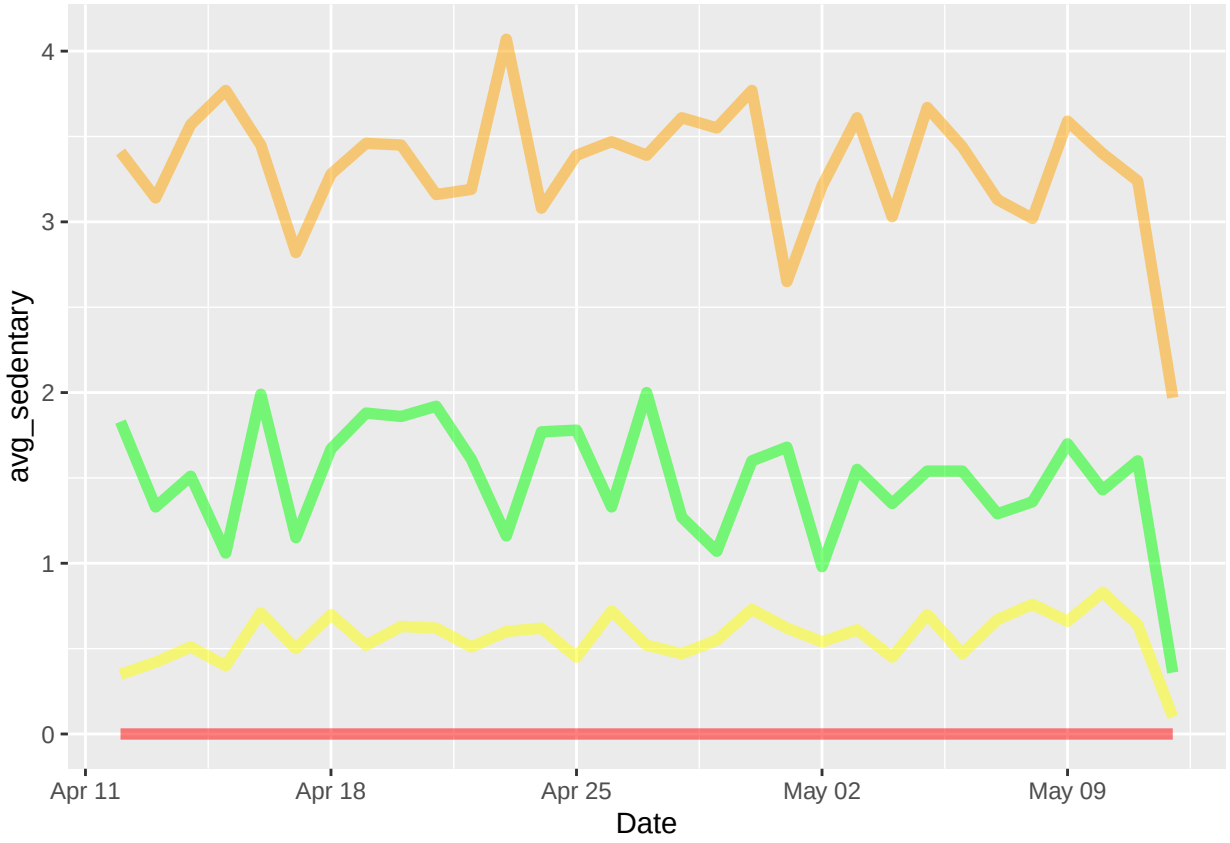


From this visual we can quickly see that most of our time is spent sedentary. Though that is not expected to change, I would like to consider subtracting sleep hours from sedentary hours to give us only waking activity.

Let's see what distance traveled based on intensity level can tell us:

```
ggplot(Intensity_distance, aes(x=Date))+
  geom_line (aes (y=avg_sedentary), color="red", alpha = 0.5, size=2)+
  geom_line (aes(y=avg_light),color="orange",alpha=0.5, size=2)+
  geom_line (aes(y=avg_moderate),color= "yellow",alpha=0.5, size=2)+
  geom_line (aes(y=avg_veryactive),color="green",alpha=0.5, size=2)
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```




```
  scale_size_continuous(range = c(1, 100))
```

```
## <ScaleContinuous>
##  Range:
##  Limits:    0 --    1
```

Suggested use for this data: We can see that people tend to cover more distance when very active compared to moderately active,but users still cover the most ground at a light level of activity! We can use this to encourage those users who aren't able to do more strenuous activities. We can also nudge users if they are spending more than the suggested time inactive, or nudge them to go a bit further on daily walks/runs ("You are 10 minutes away from beating your record! You can do it!")

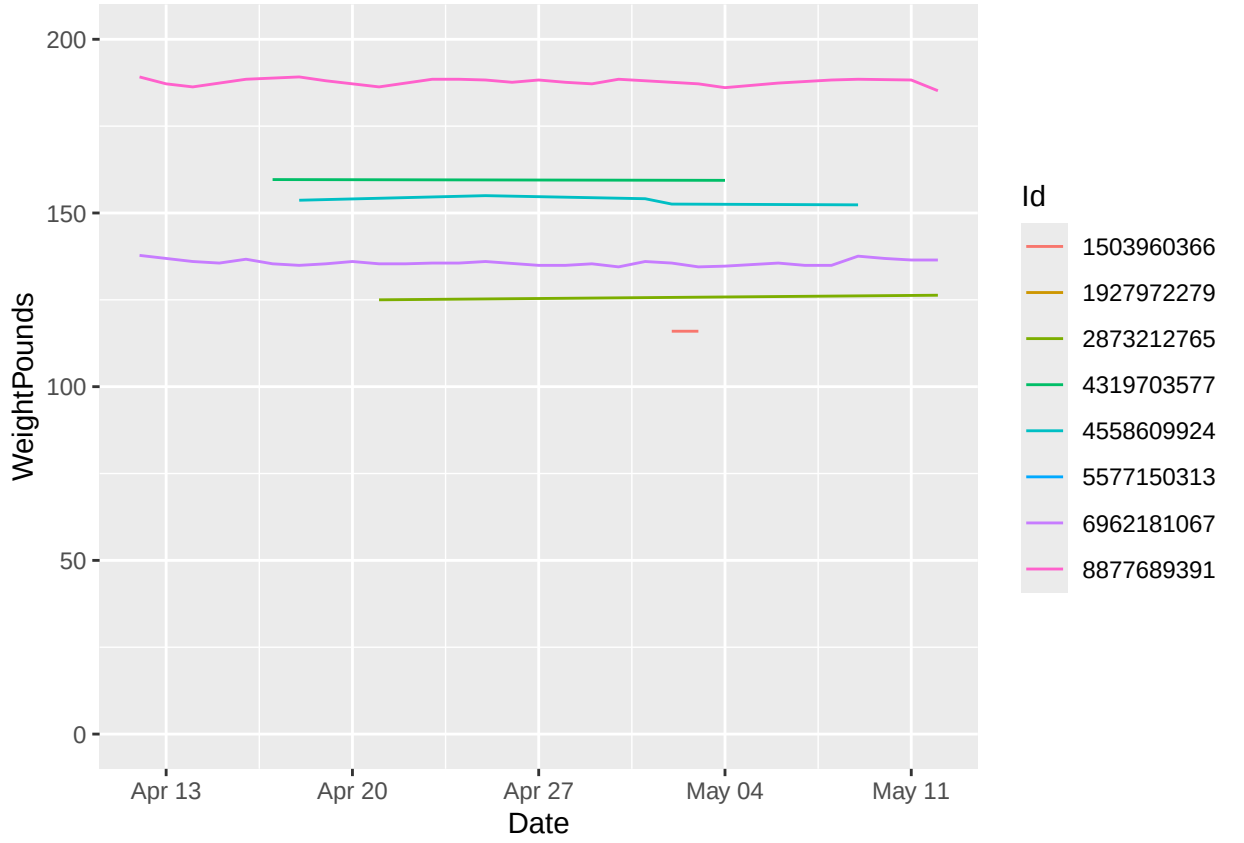
Weight

I realized earlier in the Calories section that tracking calories wasn't a main goal of these apps. Next, I want to take a look at weight. One participant had only one weigh in, we can consider it an outlier and truncate it from the graph. We will need to format the date correctly, and make sure we are considering Id a factor so we can get multiple plot lines.

```
weight$Date<-as.Date(weight$Date, format ="%m/%d/%y")
weight$Id <- as.factor(weight$Id)
```

```
ggplot(weight, aes(x = Date, y = WeightPounds, color = Id)) +
  geom_line() +
  ylim(0,200)
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
```




```
  labs(title = "Weight over Time",
       x = "Date",
       y = "Weight (Pounds)") +
  theme_minimal()
```

```
## NULL
```

Suggested use for this data: While tracking weight can be helpful if members are actively trying to lose weight, these customers do not seem to be losing weight and that may not be their focus. Given that a minority of our data set had weigh ins, and none of those had significant weight loss, fitbits main users may not be trying to lose weight. Bellabeat may want to cross reference with other platforms, or survey our own users to see how many have tried to lose weight in the last year or are likely to try in the next year. Because weight loss and calorie tracking are a significant portion of the fitness apps market, Bellabeat may want to consider including these features. It would even offer the potential tie in of a scale that automatically updates your weight through the app.

Conclusion:

In summation, Bellabeat and the Bellabeat app have the potential to grab a good segement of the market because we currently offer several things other fitness apps do not: Mindfullness activities, Menstrual cycle tracking, stress monitoring, and hydration monitoring. Bellabeat is primed to grab a top spot in market share if we can cover the bases other fitness apps do while offering things they do not yet offer.

Bellabeat already tracks activity level and sleep, as do most other apps. We could have a more holistic approach by including steps, calories in and calories out.

The website Business of Apps states that health and fitness apps made over $4 billion in 2022, and when you include online classes, subscriptions, and hardware, the industry is over $26 billion in the same year. The highest percentage of market share went to Apple fitness and Leap, with around 25% each.

With the rise of obesity, the World Health Organization has recently given a shout out to the help that MobileHealth apps give the average consumer in monitoring their own health.

Bellabeat should claim our portion of the market segment, but as a holistic business, I would like to offer a more holistic approach:

According to a study published oline, a selected group of experts rated 50 health apps, rating them in a dozen areas. They chose 7 winners in 7 different categories: Strength training, Weight loss, Mental Health, Meditation, Sleep, Hydration, and Running/Walking. There were no apps that won across two different categories.

While most apps are trying to corner a single market segment, Bellabeat can advertise to those who want to take control over multiple areas of their lives, and are tired of having to manage separate apps. Bellabeat could advertise to potential customers in a way that focuses on impowering across different areas of life:

“Take back control of your life: Why decide what to focus on when Bellabeat can help you focus on everything!”

“You’re expected to do it all, so why shouldn’t we?”

“Weather your goal is to activate or meditate, Bellabeat is here to help!”