Bellabeat Project

Ciby Lin

2022-09-28

Scenario

Bellabeat is a high-tech company specializes in health-focused products for women. Bellabeat have been sucessful and want to measure their potential to become a larger player in the global smart device market against other non-Bellabeat products. Urška Sršen is the cofounder and Chief Creative Officer of Bellabeat. We will be analyzing smart device data to form new insights on how consumers are using their smart devices. The insights discovered will then help guide marketing strategy for the company.

Ask

Business Task: Form productive marketing strategies by examing Fitbit device user behavior through different aspects to help Bellabeat become a global player in the high-tech industry.

Primary stakeholder(s):Urška Sršen, Sando Mur, and executive team members.

Secondary stakeholder(s): Bellabeat Marketing Team.

- Here are the questions we are answering:

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

Data source:

1. 30 participants FitBit Fitness Tracker Data from Mobius: https://www.kaggle.com/arashnic/fitbit (https://www.kaggle.com/arashnic/fitbit)
2. Fitness Trackers Products Ecommerce Dataset: https://www.kaggle.com/datasets/devsubhash/fitness-trackers-products-ecommerce (https://www.kaggle.com/datasets/devsubhash/fitness-trackers-products-ecommerce)

Data validation:

Reliable: The data are collected by surveying 30 eligible FitBit users through Amazon Mechanical Turk. Each participant consent to the collection of their personal tracker data and was de-identified with an unique study ID. The Fitness Trakcers Product Ecommerce Dataset has been collected from an e-commerce website (Flipkart) using webscraping technique in the Indian market. The dataset includes 565 samples with 11 attributes.

Original: Both datasets are secondary source data, since results are directly reported by the users and edited by the publishers of the datasets.

Comprehensive: Each column has a label that gives the content in the row. Dates were given in the conventional format and every value were given in metrics that are reasonable and familiar to the analyst.

Cited: Both dataset were labeled with their source and collection method. However, Fitness Trackers Products Ecommerce Dataset did not have the information on when it was collected.

Current: The Fitbit Fitness Tracker Data were collected between 03.12.2016-05.12.2016. Missing dates from Fitness Trackers Products Ecommerce Dataset.

Data limitation:

- FitBit Fitness Tracker Data has only 30 participants in total. It fulfills the central limit theorom general rule of n>=30 but a larger sample size is preferred and if time allows we could survey more users.
- Not all 30 users recorded or self-reported for WeightLog, Sleep, Steps, and Calories dataset, which makes the sample size even smaller and could cause bias.
- For the users recorded in WeightLog, 5 users manually entered their weight and 3 recorded via a connected wifi device.
- The samples in the Fitness Trackers Products Ecommerce Dataset only from Indian and it failed to recognize the other countries in the world. If allowed, more dataset from different countries are preferred.
- Both dataset were secondary source data, thus potential errors could be introduced by the editor/publisher of the dataset.

Process

Examing Daily Steps per Day

Transformed ActivityDay from character to date while formatting into MM/DD/YY.

```
head(as.Date(steps_day$ActivityDay,format="%m%d%y"))
```

```
## [1] "2016-04-12" "2016-04-13" "2016-04-14" "2016-04-15" "2016-04-16"
## [6] "2016-04-17"
```

```
data<-head(steps_day, 10)
knitr::kable(data,
 caption = "StepTotal Table")
```

StepTotal Table

ActivityDay	Id	StepTotal
2016-04-12	1503960366	13162
2016-04-13	1503960366	10735
2016-04-14	1503960366	10460
2016-04-15	1503960366	9762
2016-04-16	1503960366	12669
2016-04-17	1503960366	9705
2016-04-18	1503960366	13019
2016-04-19	1503960366	15506
2016-04-20	1503960366	10544
2016-04-21	1503960366	9819

Ordering ActivityDay in ascending order so the dates and resulting data are in chronological order when plotted. Then we grouped the StepTotal by dates and find the avergae steps per day while keeping the arrnagement in ascending order.

```
new_steps_day <- steps_day %>% group_by(ActivityDay) %>% summarize(mean_step = mean(StepTotal))
print(new_steps_day) %>%  arrange(ActivityDay)
```

```
## # A tibble: 31 × 2
##    ActivityDay mean_step
##    <date>          <dbl>
##  1 2016-04-12      8237.
##  2 2016-04-13      7199.
##  3 2016-04-14      7744.
##  4 2016-04-15      7534.
##  5 2016-04-16      8679.
##  6 2016-04-17      6409.
##  7 2016-04-18      7897.
##  8 2016-04-19      8049.
##  9 2016-04-20      8163.
## 10 2016-04-21      8244.
## # … with 21 more rows
## # ℹ Use `print(n = ...)` to see more rows
```

```
## # A tibble: 31 × 2
##    ActivityDay mean_step
##    <date>          <dbl>
##  1 2016-04-12      8237.
##  2 2016-04-13      7199.
##  3 2016-04-14      7744.
##  4 2016-04-15      7534.
##  5 2016-04-16      8679.
##  6 2016-04-17      6409.
##  7 2016-04-18      7897.
##  8 2016-04-19      8049.
##  9 2016-04-20      8163.
## 10 2016-04-21      8244.
## # … with 21 more rows
## # ℹ Use `print(n = ...)` to see more rows
```

```
data<-head(new_steps_day, 10)
knitr::kable(data,
 caption = "Average Daily Steps Table")
```

Average Daily Steps Table

ActivityDay	mean_step
2016-04-12	8236.848
2016-04-13	7198.727
2016-04-14	7743.576
2016-04-15	7533.848
2016-04-16	8679.156
2016-04-17	6409.250
2016-04-18	7896.969
2016-04-19	8048.656
2016-04-20	8162.969
2016-04-21	8243.594

Since there are duplicated ActivityDates in the long table, we group StepTotal by same ActivityDay and find the average of the steps. StepTotal were not grouped by duplicated Ids because the results need to include and combine the average step counts among all 33 participants (checked with n_distinct()). In the “new_steps_day” data frame, the mean/average of StepTotal of all participants grouped by the same dates were founded and put into Daily Steps Table by processing the data frame with a combination of summarise, group_by, and mean functions.

Examing Calories Dataset

ActivityDay was transformed from character to date and formatted in MM/DD/YY format. Then only the zeros were removed, no NAs were detected in the dataset to remove. After removing the zeros we find the summary of the dataset and found the minimun being 52 calories which is much smaller than the first quartile 1834, so we removed any values that are smaller than the first quartile. Assuming some participants forgot to report all of their consumed calories per day and caused the two-digit values. Lastly, the dates were put in an ascending order so it’s prepared for visualization.

```
head(as.Date(calories$ActivityDay,format="%m%d%y"))
```

```
## [1] "2016-04-12" "2016-04-13" "2016-04-14" "2016-04-15" "2016-04-16"
## [6] "2016-04-17"
```

```
calories_cleaned <- calories %>% filter(Calories !=0, Calories > 1834) %>%
  arrange(ActivityDay)
```

Average Calories were grouped by the same dates first then devided among the dates and fitlered by the first quartile to avoid the huge dip at the end of the graph caused by any outliers, then we turned the results into the Calories table.

```
new_calories<- calories_cleaned %>% group_by(ActivityDay) %>%
  summarize(mean_calorie=mean(Calories)) %>% filter(mean_calorie > 2285)

knitr::kable(head(new_calories, 10),
 caption = "Average Calories")
```

Average Calories

ActivityDay	mean_calorie
2016-04-12	2545.286

ActivityDay	mean_calorie
2016-04-13	2503.240
2016-04-14	2642.083
2016-04-15	2636.625
2016-04-16	2647.708
2016-04-17	2539.364
2016-04-18	2558.958
2016-04-19	2577.360
2016-04-20	2619.480
2016-04-21	2654.760

Examing Acitivty Intensity Dataset

Preparing data to create two types of visualizations: layered line chart and pie chart. use Activity Minutes instead of Distance to investigate because minute has higher temporal resolution than distance and help avoid ambiguity.

To find the average Activity Minutes of each intensity:

```
new_intensity <- intensity %>% group_by(ActivityDay) %>% summarize(mean_sit_min=mean(SedentaryMinutes),
  mean_lit_min=mean(LightlyActiveMinutes), mean_fair_min=mean(FairlyActiveMinutes),
  mean_act_min=mean(VeryActiveMinutes))
```

```
data <- head(new_intensity,10)
knitr::kable(data,
 caption = "Average Active Minutes for each Intensity")
```

Average Active Minutes for each Intensity

ActivityDay	mean_sit_min	mean_lit_min	mean_fair_min	mean_act_min
2016-04-12	1026.2121	199.0000	7.848485	22.30303
2016-04-13	1021.7879	181.7576	10.575758	20.33333
2016-04-14	1010.0303	201.0000	12.393939	20.93939
2016-04-15	961.0606	213.8485	9.878788	19.18182
2016-04-16	1002.6562	193.8125	15.125000	27.84375
2016-04-17	1049.9688	165.3438	11.843750	18.90625
2016-04-18	1061.2188	188.2812	16.125000	24.40625
2016-04-19	1003.9375	201.9062	13.781250	23.96875
2016-04-20	974.1250	203.5938	18.750000	24.18750
2016-04-21	1031.8750	182.6562	14.937500	26.84375

Examing WeightLog Dataset

Removed one outlier of BMI because there were 67 measurements, which is still enough for us to see a trend in the dataset after removing one outlier. We removed the outlier that is greater than the third quartile (37.2).

```
quartile<- weight %>% summarize(mean_bmi=mean(BMI))*1.5
weight<-filter(weight, BMI< 30)
```

Cleaned and simplified the data by grouping by date and find the average weights and BMI from the dataset, then turn the results into a table:

```
new_weight <- weight %>% group_by(DATE) %>% summarize(mean_kg=mean(WeightKg),
mean_pounds=mean(WeightPds), mean_BMI=mean(BMI))
data <- head(new_weight, 10)
knitr::kable(data,
 caption = "Average Weight and BMI")
```

Average Weight and BMI

DATE	mean_kg	mean_pounds	mean_BMI
2016-04-12	74.15000	163.4728	25.03500
2016-04-13	73.50000	162.0398	24.82500
2016-04-14	73.10000	161.1579	24.70500
2016-04-15	61.50000	135.5843	24.00000
2016-04-16	73.75000	162.5909	24.90000
2016-04-17	74.83333	164.9793	26.47000
2016-04-18	72.23333	159.2472	25.60667
2016-04-19	73.35000	161.7091	24.74500
2016-04-20	73.30000	161.5988	24.75500
2016-04-21	67.53333	148.8855	23.56667

Examning DailySleep Dataset

sleep_day dataframe was grouped by same dates and used the mean() function to find the average TotalMinutesSleep and avergae TotalTimeInBed, then arranged the results into Average Minutes of Sleep table.

```
new_sleep <- sleep_day %>%
  group_by(DATE) %>%
  summarize(mean_asleep=mean(TotalMinutesSleep),
            mean_inbed=mean(TotalTimeInBed))
data<- head(new_sleep, 10)
  knitr::kable(data, caption="Average Minutes of Sleep")
```

Average Minutes of Sleep

DATE	mean_asleep	mean_inbed
2016-04-12	441.9231	479.6923
2016-04-13	430.4286	471.8571
2016-04-14	445.2308	480.2308
2016-04-15	427.4706	476.3529
2016-04-16	391.7143	433.0000
2016-04-17	464.0833	509.1667
2016-04-18	419.9000	455.9000
2016-04-19	409.0714	451.5714
2016-04-20	446.2667	476.7333
2016-04-21	376.0000	409.3333

Examing Fitness Tracker Dataset

Use is.na to filter out the missing values among the Ratings:

```
fit_tracker %>% filter(!is.na(Rating__Out_of_5_))
```

```
## # A tibble: 554 × 8
##    Brand_Name Device_type Model_Name       Color Selli…¹ Origi…² Ratin…³ Strap…⁴
##    <chr>      <chr>       <chr>            <chr>   <dbl>   <dbl>   <dbl> <chr>
##  1 APPLE      Smartwatch  42 mm White Cer… Cloud  114900  114900     4.7 Nylon
##  2 APPLE      Smartwatch  Series 3 GPS + … Black  122090  122090     4.6 Nylon
##  3 APPLE      Smartwatch  Series 3 GPS + … Black  122090  122090     4.6 Nylon
##  4 GARMIN     Smartwatch  Vivomove Style   Whit…   26990   31490     3.3 Nylon
##  5 Honor      Smartwatch  GS Pro           Black   13999   20999     4.4 Nylon
##  6 Honor      Smartwatch  GS Pro           Blue    17999   20999     4.4 Nylon
##  7 FitBit     Smartwatch  Versa Special E… Char…   10365   23499     4.1 Fabric
##  8 Zebronics  Smartwatch  Smarttime 200    Black    1499    2999     3.1 Rubber
##  9 FitBit     FitnessBand Flex Small       Viol…    8490    8490     3.8 Rubber
## 10 SAMSUNG    FitnessBand Galaxy Fit-e Sm… Black    2000    2590     3.8 Rubber
## # … with 544 more rows, and abbreviated variable names ¹Selling_Price,
## #   ²Original_Price, ³Rating__Out_of_5_, ⁴Strap_Material
## # ℹ Use `print(n = ...)` to see more rows
```

```
data <- head(fit_tracker, 10)
knitr::kable(data,
 caption = "Fitness Tracker Ecommerce Table")
```

Fitness Tracker Ecommerce Table

Brand_Name	Device_type	Model_Name	Color	Selling_Price	Original_Price	Rating_*Out_of_5*	Strap_Material
APPLE	Smartwatch	42 mm White Ceramic Case with Cloud Sport	Cloud	114900	114900	4.7	Nylon
APPLE	Smartwatch	Series 3 GPS + Cellular- 42 mm Gray Ceramic Case	Black	122090	122090	4.6	Nylon
APPLE	Smartwatch	Series 3 GPS + Cellular- 42 mm White Ceramic Case	Black	122090	122090	4.6	Nylon
GARMIN	Smartwatch	Vivomove Style	White, Pink	26990	31490	3.3	Nylon
Honor	Smartwatch	GS Pro	Black	13999	20999	4.4	Nylon
Honor	Smartwatch	GS Pro	Blue	17999	20999	4.4	Nylon
GARMIN	Smartwatch	Fenix 6	Black, Red, Orange	76990	86690	NA	Nylon
GARMIN	Smartwatch	Fenix 6 Pro Solar	Grey, Black	72990	88490	NA	Nylon
GARMIN	Smartwatch	Fenix 6X	Black, Orange, Red	79990	88490	NA	Nylon
GARMIN	Smartwatch	Fenix 6	Grey, Blue	77990	86690	NA	Nylon

Create 5 dataframe for each level of rating, which is out of 5 and exam the behavior in the top two levels to find the distinct characteristics of high-performing products:

```
lvl1<- filter(fit_tracker, Rating__Out_of_5_ %in% (0:1))
lvl2<- filter(fit_tracker, Rating__Out_of_5_ %in% (1:2))
lvl3<- filter(fit_tracker, Rating__Out_of_5_ %in% (2:3))
lvl4<- filter(fit_tracker, Rating__Out_of_5_ %in% (3:4))
lvl5<- filter(fit_tracker, Rating__Out_of_5_ %in% (4:5))
```

```
new_fit_tracker<-fit_tracker %>% group_by(Rating__Out_of_5_) %>%
  summarize(Selling_Price=mean(Selling_Price),
            Original_Price=mean(Original_Price))
data <- head(new_fit_tracker, 10)
knitr::kable(data,
 caption = "Fitness Tracker Rating vs Price")
```

Fitness Tracker Rating vs Price

Rating_*Out_of_5*	Selling_Price	Original_Price
2.0	16990.000	16990.00
2.3	19999.000	22550.00
2.4	5846.000	5999.00

Rating_*Out_of_5*	Selling_Price	Original_Price
2.5	799.000	3199.00
2.8	12497.000	12497.00
2.9	3297.000	4199.00
3.0	11244.750	15367.50
3.1	5873.333	6615.00
3.2	26028.500	31120.50
3.3	15603.875	17880.88

Analyze

Examing Daily Steps Dataset

Using data frame containing dates and step counts of each participant each day. We can exame wether they use fitness trackers to increase their daily activity by increasing their daily steps.

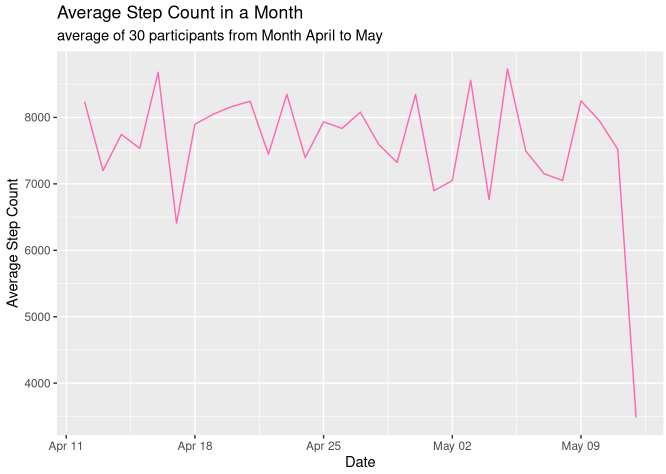
Steps Summary

The mean of the average steo is 7592 steps per day and it is closer to the maximun step 8731 than minimun step 3482. This correlates to the slight increase we witness in the plot below.

```
summary(new_steps_day)
```

```
##   ActivityDay           mean_step
##  Min.   :2016-04-12   Min.   :3482
##  1st Qu.:2016-04-19   1st Qu.:7260
##  Median :2016-04-27   Median :7744
##  Mean   :2016-04-27   Mean   :7592
##  3rd Qu.:2016-05-04   3rd Qu.:8200
##  Max.   :2016-05-12   Max.   :8731
```

```
ggplot(new_steps_day, aes(ActivityDay, mean_step, group=1))+
  geom_line(color="hot pink")+
  labs(title="Average Step Count in a Month",subtitle=
         "average of 30 participants from Month April to May",
       x="Date", y= "Average Step Count")
```



In the line plot above, there is no obvious trend, increasing, or decreasing in step counts from April 13th till May 12th. However, there is a hige dip/reduction in avergae step at the end. The data framed was investigated and cleaned. We hypothesize that some participants might have forgot to charge their watch or lost interest/habit of wearing the fitness tracker. We will further investigate the reason behind the dip. The average steps per day for aevrage American is between 3,000 to 4,000 according to Mayo Clinic (https://www.mayoclinic.org/healthy-lifestyle/fitness/in-depth/10000-steps/art-20317391#:~:text=The%20average%20American%20walks%203%2C000,a%20day%20every%20two%20weeks.). The trend line is fluctuation mostly ranging from 7,000 to 8,000 steps per day on average. We do see a higher than average steps per day with participants wearing the FitBit device. However, there could be experimenter effect as a confounding variable, so replication of the survey is needed. We could assume the device did help the participant walk more for now.

Examing Calories Dataset

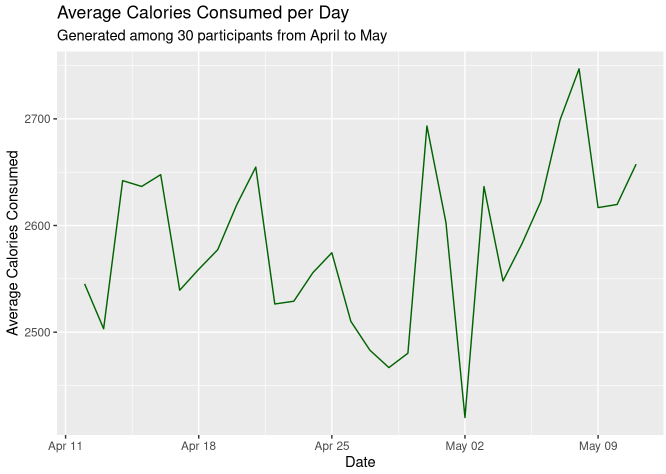
This section is to investigate whether the participants used the fitness trackers to reduce their calorie intake and maintain a healthier diet. #### Calories Summary Since the mean and median are more than 100 calories apart will be using the median calorie 2331 as the representation of the peak of the distribution curve. The median calories per day is closer to the first quartile 2285 than the third quartile 2383, thus we would be expecting a slight right skewed distribution if we plotted one, which means most of the calories are in the reasonable range. According to WHO (https://www.who.int/news-room/fact-sheets/detail/healthy-diet#:~:text=For%20adults&text=Less%20than%2010%25%20of%20total%20energy%20intake%20from%20free%20sugars,additional%20health%20benefits%20(7).), a healthy person should consume approximately 2000 calories per day. According to Mayo Clinic (https://www.mayoclinic.org/healthy-lifestyle/weight-loss/in-depth/calories/art-20048065#:~:text=In%20general%2C%20if%20you%20cut,your%20gender%20and%20activity%20level.), one needs to cut 500 calories per day from normal diet to lose weight.

```
summary(new_calories)
```

```
##   ActivityDay          mean_calorie
##  Min.   :2016-04-12   Min.   :2420
##  1st Qu.:2016-04-19   1st Qu.:2532
##  Median :2016-04-26   Median :2580
##  Mean   :2016-04-26   Mean   :2583
##  3rd Qu.:2016-05-03   3rd Qu.:2637
##  Max.   :2016-05-11   Max.   :2747
```

The calories ranges from 2287 to 2454 which is in the accepted range for a healthy person. The outlier minimun value 1259 and other outliers less than 2285 were removed, because even if the participants wanted to lose weight at the end of the survey period, cutting approximately 1000 calories is not sustainable. We see a drop in average calories consumed around April 11th and April 24th and a huge growth around May 2nd. Overall there is not correlation between increasing days and calories consumsion. Thus, we can hypothesize that the users did use Fitbit to watch their calories but not for weight loss purposes. We can further investigate in the WeightLog section below.

```
ggplot(new_calories, aes(ActivityDay, mean_calorie))+
  geom_line(color="dark green")+
  labs(title="Average Calories Consumed per Day",
       subtitle="Generated among 30 participants from April to May",
       x="Date", y="Average Calories Consumed")
```



Examing Acitivty Intensity Dataset

Intensity Summary

Average sedentary minutes have the highest average of 986.4 min, while lightly active minutes have an average of 191.57 min, fairly active minutes is 13.5 min, with very active minutes average being 20.96 min.

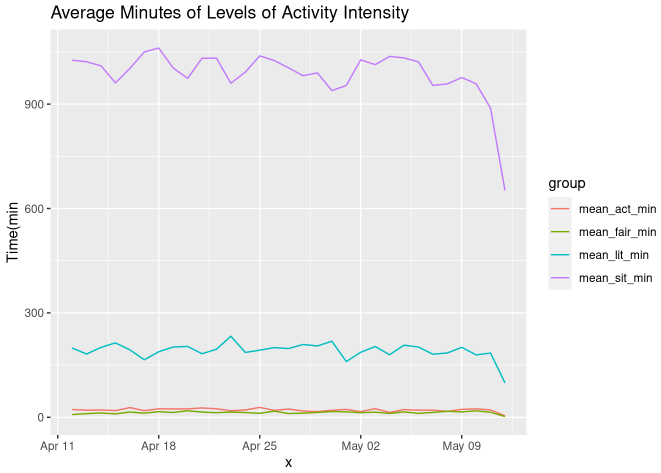
```
summary(new_intensity)
```

```
##   ActivityDay          mean_sit_min     mean_lit_min    mean_fair_min
##  Min.   :2016-04-12   Min.   : 652.0   Min.   : 98.81   Min.   : 2.143
##  1st Qu.:2016-04-19   1st Qu.: 960.5   1st Qu.:183.60   1st Qu.:11.594
##  Median :2016-04-27   Median :1003.9   Median :195.53   Median :14.000
##  Mean   :2016-04-27   Mean   : 986.4   Mean   :191.57   Mean   :13.522
##  3rd Qu.:2016-05-04   3rd Qu.:1026.8   3rd Qu.:202.64   3rd Qu.:15.465
##  Max.   :2016-05-12   Max.   :1061.2   Max.   :232.91   Max.   :18.750
##   mean_act_min
##  Min.   : 4.19
##  1st Qu.:19.04
##  Median :21.03
##  Mean   :20.96
##  3rd Qu.:24.08
##  Max.   :28.41
```

```
data_ggp <- data.frame(x = new_intensity$ActivityDay,
                       y = c(new_intensity$mean_sit_min,
                             new_intensity$mean_lit_min,
                             new_intensity$mean_fair_min,
                             new_intensity$mean_act_min),
group = c(rep("mean_sit_min", nrow(new_intensity)),
rep("mean_lit_min", nrow(new_intensity)),
rep("mean_fair_min", nrow(new_intensity)),
rep("mean_act_min", nrow(new_intensity))))
```

Below is the graph of all average minutes. All three levels of activity are all steady and there is no sign of increase or decrease in average minutes, nor were there any crossovers of trend lines.

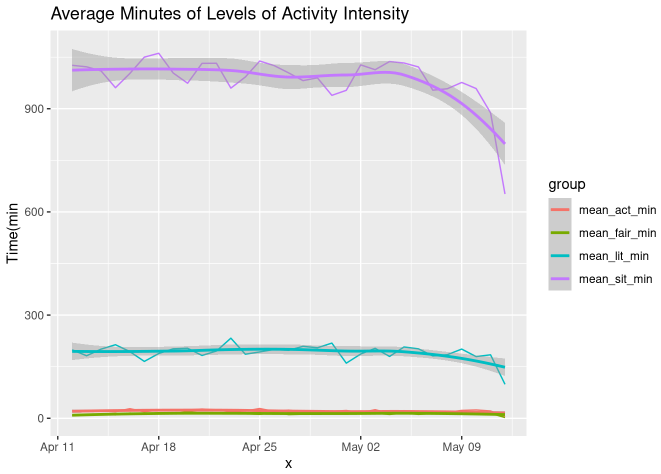
```
ggplot(data_ggp, aes(x,y,col=group)) + geom_line()+labs(title="Average Minutes of Levels of Activity Intensity",
X="Date", y="Time(min")
```



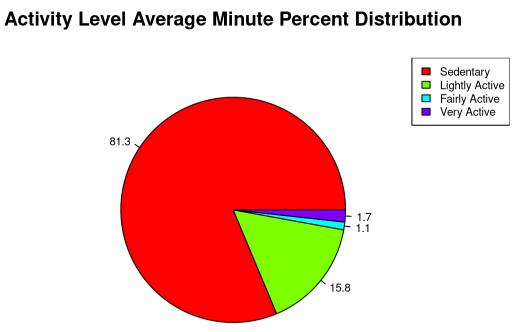
Using geom_smooth() function we obtained curves that shows the correlation of each plot. We see a strong correlation between the plots among the same level and little to no fluctuation in the trend lines. From the observation, we can hypothesize that the Fitbit device were slightly effective in reducing the sedentary minutes while having no effect on other three higher activity levels.

```
ggplot(data_ggp, aes(x,y,col=group)) + geom_line()+geom_smooth()+labs(title="Average Minutes of Levels of Activit
y Intensity", X="Date", y="Time(min")
```

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




```
act_lvl <- c(986, 192, 13.5, 21)
piepercent <- round (100* (act_lvl/sum(act_lvl)),1 )
pie_v2 <- pie(act_lvl, labels=piepercent, cex=0.7, main="Activity Level Average Minute Percent Distribution", col
=rainbow(length(act_lvl)) )
legend("topright", c("Sedentary","Lightly Active", "Fairly Active", "Very Active"),
       cex = 0.7, fill = rainbow(length(act_lvl)))
```



Above is a pie chart further visualize the distribution of different in active level, and we can see that on average among the whole time wearign the device the users are 81.3% sedentary, 15.8% lightly active, 1.1% fairly active, and 1.7% very active. One thing to notice is the percent of very active is 0.6% more than fairly active minutes and it was made more obvious by the pie chart. However, since there is no crossover nor fluctuation in the two bottom trend lines in the line plot. We can't assume the device have any effect on increasing the active minutes.

Examing WeightLog Dataset

WeightLog Summary

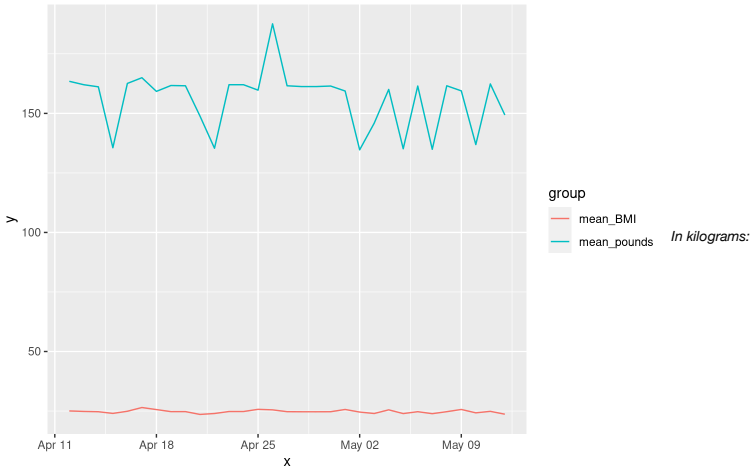
The mean_BMI column of the dataset has a mean BMI of 25.00. According to CDC (https://www.cdc.gov/healthyweight/assessing/bmi/adult_bmi/index.html) for male and female over the age of 20, the range for healthy BMI is between 18.5 to 24.9. The minimum average BMI of the dataset is 23.7 and the average maximum BMI is 32.40, which are both on a larger side of the scale.

```
summary(new_weight)
```

```
##       DATE              mean_kg       mean_pounds       mean_BMI
##  Min.   :2016-04-12   Min.   :61.10   Min.   :134.7   Min.   :23.57
##  1st Qu.:2016-04-19   1st Qu.:67.63   1st Qu.:149.1   1st Qu.:24.40
##  Median :2016-04-27   Median :73.15   Median :161.3   Median :24.72
##  Mean   :2016-04-27   Mean   :70.75   Mean   :156.0   Mean   :24.76
##  3rd Qu.:2016-05-04   3rd Qu.:73.42   3rd Qu.:161.9   3rd Qu.:24.97
##  Max.   :2016-05-12   Max.   :85.10   Max.   :187.6   Max.   :26.47
```

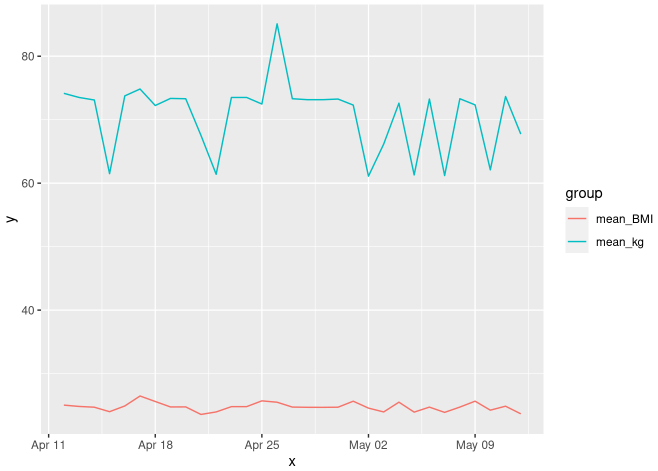
In the line graph below, we can see the top line being the change in weight through out the survey period and the bottom red line is the change in BMI. There is a steady decrease in weight and BMI through out the time with a slight increase in weight in the middle. Same trend is spotted for both plots. *In Pounds:*

```
data_ggplot <- data.frame(x = new_weight$DATE,
                      y = c(new_weight$mean_pounds,
                            new_weight$mean_BMI),
                      group = c(rep("mean_pounds", nrow(new_weight)),
                                rep("mean_BMI", nrow(new_weight))))
ggplot(data_ggplot, aes(x,y,col=group)) + geom_line()
```



In kilograms:

```
data_ggplot1 <- data.frame(x = new_weight$DATE,
                       y = c(new_weight$mean_kg,
                             new_weight$mean_BMI),
                       group = c(rep("mean_kg", nrow(new_weight)),
                                 rep("mean_BMI", nrow(new_weight))))
ggplot(data_ggplot1, aes(x,y,col=group)) + geom_line()
```

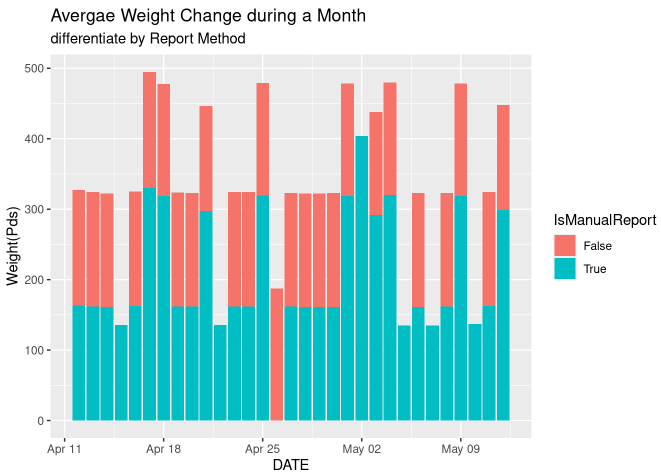


bar graphs:

```
bar_weight <- weight %>% group_by(DATE) %>% summarize(mean_kg=mean(WeightKg),
mean_pounds=mean(WeightPds), mean_BMI=mean(BMI), IsManualReport)
```

```
## `summarise()` has grouped output by 'DATE'. You can override using the
## `.groups` argument.
```

```
bar_weight %>% ggplot()+geom_col( aes(x=DATE, y=mean_pounds, fill=IsManualReport))+
  labs(title="Avergae Weight Change during a Month", subtitle="differentiate by Report Method", y="Weight(Pds)")
```




```
sum(weight$IsManualReport=="False")
```

```
## [1] 25
```

```
sum(weight$IsManualReport=="True")
```

```
## [1] 41
```

Above is a bar graph showing the change in weight with each bar distinguished by the amount of results being self-reported or wifi-reported. There are 41 self-reported and 25 wifi-reported results. We can assume the self-reporting participants would be more conscious of the change in their weight and have more drive to be healthier and reduce their BMI. However, having two different reporting method can create a confounding variable which could cause the result not applicable to the whole population.

Examning DailySleep Dataset

Sleep Summary

The difference bewtween the minimum, mean, and maximum time in bed and minutes asleep are approximately the same. The participants spent an average of 458.8 minutes in bed and 420.1 minutes asleep. The average time asleep is more than the average american time asleep, 408 minutes.

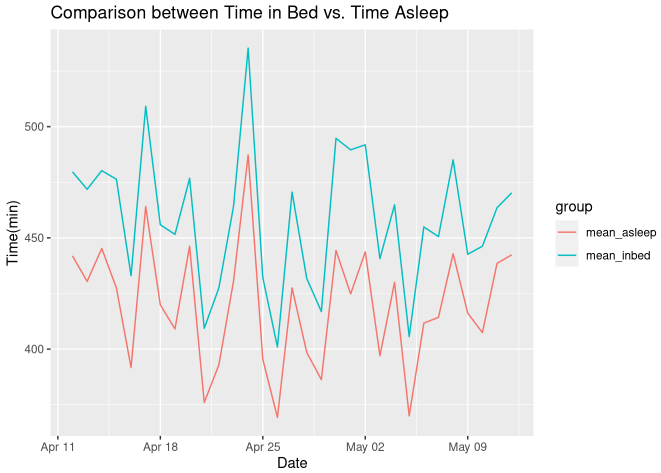
```
summary(new_sleep)
```

```
##       DATE             mean_asleep      mean_inbed
##  Min.   :2016-04-12   Min.   :369.3   Min.   :400.9
##  1st Qu.:2016-04-19   1st Qu.:397.8   1st Qu.:436.8
##  Median :2016-04-27   Median :424.8   Median :463.5
##  Mean   :2016-04-27   Mean   :420.1   Mean   :458.8
##  3rd Qu.:2016-05-04   3rd Qu.:442.1   3rd Qu.:478.2
##  Max.   :2016-05-12   Max.   :487.3   Max.   :535.3
##  NA's   :1            NA's   :1       NA's   :1
```

Below is the line graph showing the relationship between the time in bead and the time asleep. We can see that the gap between the two lines have been fluctuating but not closing at the end of the survey period, which means the sleep quality was not increase while wearing the FitBit device, but the time asleep is overall higher than the average Americans sleeping 6.8 hours (https://news.gallup.com/poll/166553/less-recommended-amount-sleep.aspx).

```
data_ggp <- data.frame(x = new_sleep$DATE,
                       y = c(new_sleep$mean_asleep,
                             new_sleep$mean_inbed),
                       group = c(rep("mean_asleep", nrow(new_sleep)),
                                 rep("mean_inbed", nrow(new_sleep))))
ggplot(data_ggp, aes(x,y,col=group)) + geom_line() +
  labs(title="Comparison between Time in Bed vs. Time Asleep", x="Date", y="Time(min)")
```

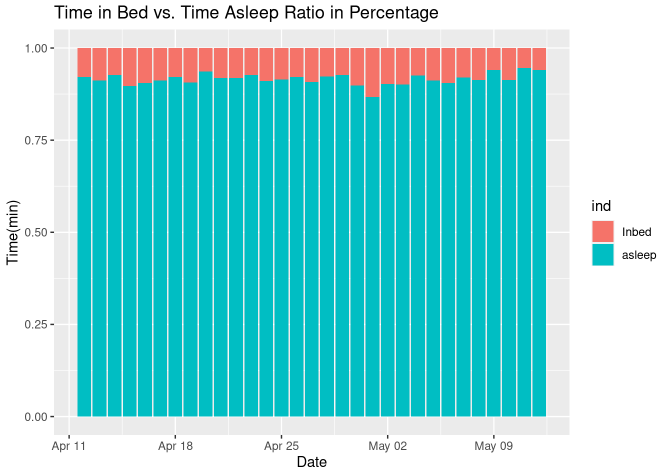
```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```



Below is a stacked bar graph showing the ratio of time asleep out of the time in bed. We can see that most of the time is spent asleep, so we can also assume that the participants have sufficient sleep quality on average and would not have the need for better sleep quality. This requires further investigation on the sleep quality change of other FitBit users.

```
percent_sleep<- data.frame(new_sleep$DATE, asleep=c(new_sleep$mean_asleep/new_sleep$mean_inbed),
                           sleep=c(new_sleep$mean_inbed-new_sleep$mean_asleep)/new_sleep$mean_inbed)
data <- data.frame(col1=c(percent_sleep$new_sleep.DATE),
                   Inbed=c(percent_sleep$sleep),
                   asleep=c(percent_sleep$asleep))
data_mod1 <- cbind(data[1], stack(data[2:3]))
data_mod1 %>% ggplot()+ geom_col(aes(x=col1, y=values, fill=ind))+
  labs(title="Time in Bed vs. Time Asleep Ratio in Percentage", x="Date", y="Time(min)")
```

```
## Warning: Removed 2 rows containing missing values (position_stack).
```



Examing Fitness Tracker Dataset

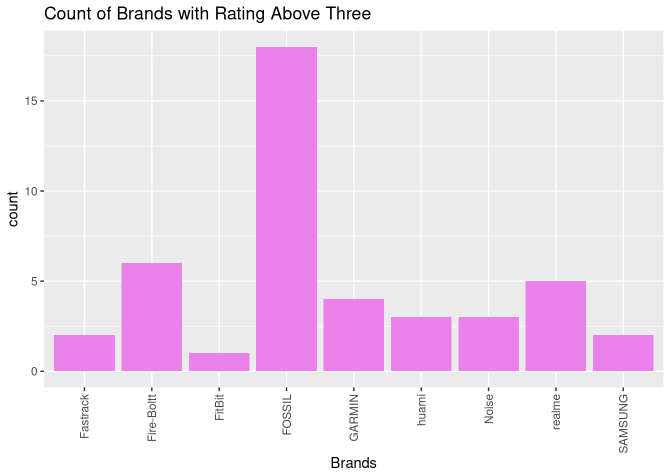
Fitness Tracker Ecommerce Summary

The two types of prices are both in Indian rupee. One US dollar is equivalent to approximately 80 rupees. Since the medians and means are quite far apart, we will be using the medians as representations of mean. The average selling price is approximately $187, and the original price is approximately $237.

```
fit_tracker1<-select(fit_tracker, Selling_Price, Original_Price, Rating__Out_of_5_)
summary(fit_tracker1)
```

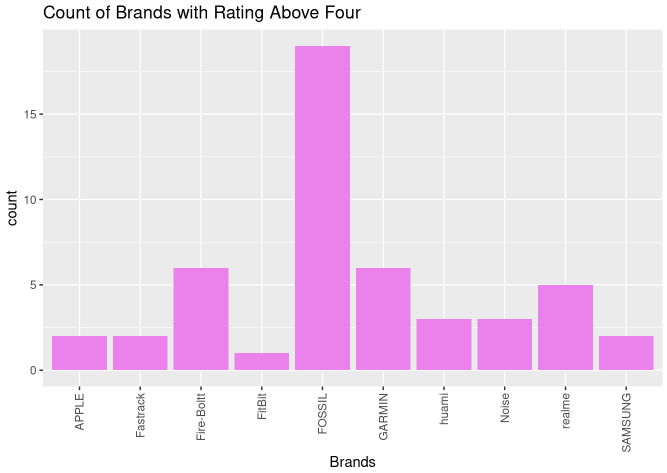
```
##  Selling_Price    Original_Price   Rating__Out_of_5_
##  Min.   :   799   Min.   :  1599   Min.   :2.000
##  1st Qu.:  6995   1st Qu.: 10249   1st Qu.:4.000
##  Median : 14999   Median : 18995   Median :4.200
##  Mean   : 20707   Mean   : 23978   Mean   :4.196
##  3rd Qu.: 27468   3rd Qu.: 31417   3rd Qu.:4.500
##  Max.   :122090   Max.   :122090   Max.   :5.000
##                                    NA's   :56
```

```
lvl4 %>% ggplot()+geom_bar(aes(x=Brand_Name), fill="Violet")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(title="Count of Brands with Rating Above Three", x="Brands")
```



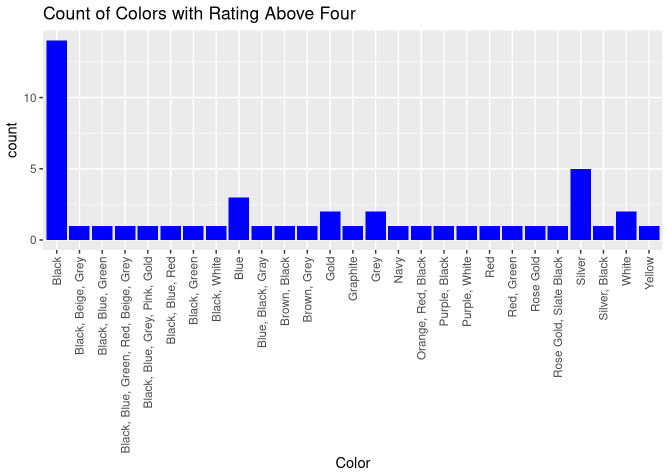
The bar chart above compares the number of each brand's device that has a rating more than three. Fossil's devices have the most ratings above 3 and less than 4, Fire-Boltt is secon and realme being the third most with high ratings.

```
lvl5 %>% ggplot()+geom_bar(aes(x=Brand_Name), fill="Violet")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(title="Count of Brands with Rating Above Four", x="Brands")
```



Above is the bar chart also comparing the amount of device having a rating over 4 and below 5. Fossil once again has the most device with ratings over 4, while Fire-Boltt and GARMIN both come in second, and realme being the third.

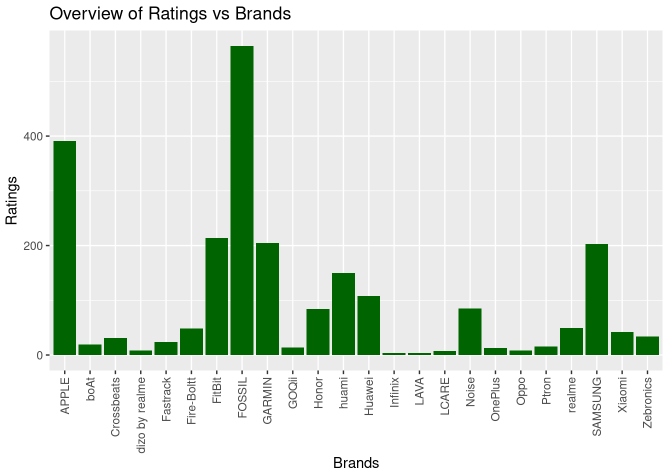
```
lvl5 %>% ggplot()+geom_bar(aes(x=Color), fill="blue")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(title="Count of Colors with Rating Above Four", x="Color")
```



Above is the bar chart showing the colors of device that have the highest ratings. Black is the most popular. with silver being second, and blue being the third most popular color. Follow by gold, grey and white.

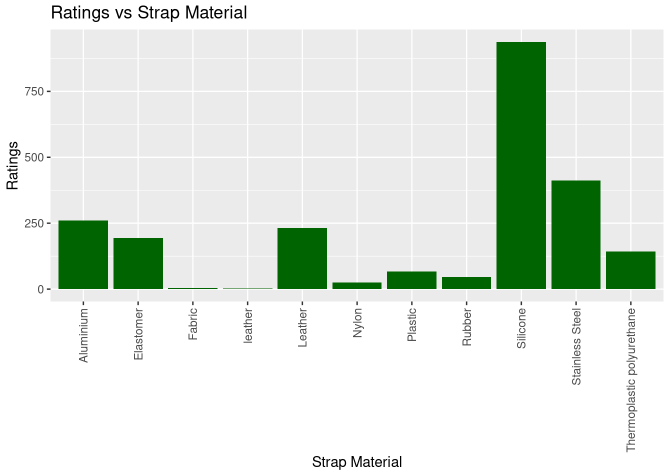
```
fit_tracker %>% ggplot()+geom_col(aes(x=Brand_Name, y=Rating__Out_of_5_), fill="dark Green")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+labs(title="Overview of Ratings vs Brands",
x="Brands", y="Ratings")
```

```
## Warning: Removed 56 rows containing missing values (position_stack).
```




```
fit_tracker %>% ggplot()+geom_col(aes(x=Strap_Material, y=Rating__Out_of_5_), fill="dark green")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+labs(title="Ratings vs Strap Material", x
="Strap Material", y="Ratings")
```

```
## Warning: Removed 56 rows containing missing values (position_stack).
```



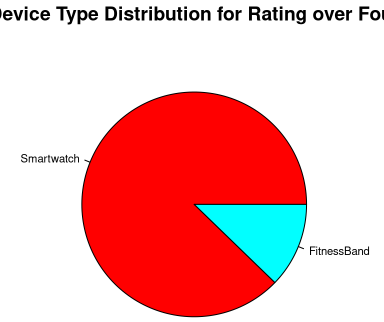
The bar chart above compares the popularity of different strap materials. The results show that silicone is the most popular, stainless steel is second, and aluminum being the third most popular.

```
pie_device <- lvl5 %>% group_by() %>% summarize(pie_watch=sum(lvl5$Device_type=='Smartwatch'), pie_band=sum(lvl5
$Device_type=='FitnessBand'))

print(pie_device)
```

```
## # A tibble: 1 × 2
##   pie_watch pie_band
##       <int>    <int>
## 1        43        6
```

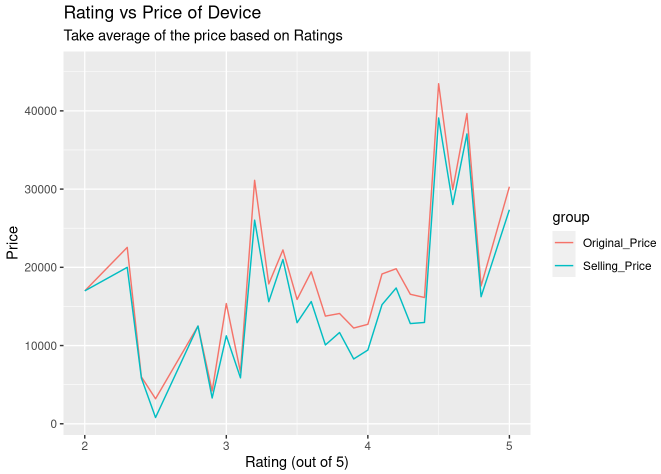
```
rating_lvl <- c( 43, 6)
labels <- c("Smartwatch", "FitnessBand")
pie_v1 <- pie(rating_lvl, labels, cex=0.7, main="Device Type Distribution for Rating over Four", col=rainbow(leng
th(rating_lvl)) )
```



Next, we compare the popularity between Smartwatch and FitnessBand. Smartwatch did show up more in the higher ratings. However, Smartwatch is the dominant device type among the two in general, thus we can't make an assumption based on this observation. More investigation in different parts of the world is needed.

```
data_ggpp <- data.frame(x = new_fit_tracker$Rating__Out_of_5_,
                        y = c(new_fit_tracker$Selling_Price,
                              new_fit_tracker$Original_Price),
                        group = c(rep("Selling_Price", nrow(new_fit_tracker)),
                                  rep("Original_Price", nrow(new_fit_tracker))))
ggplot(data_ggpp, aes(x,y,col=group)) + geom_line()+
  labs(title="Rating vs Price of Device", subtitle="Take average of the price based on Ratings", x="Rating (out o
f 5)", y="Price")
```

```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```



The geom_line plot above show is the trend between price and rating. We inclued the original price and selling price and from the plot we can tell the gap between the original price and selling price are not that big for most ratings beside ratings from 3.5 to 4.5. The bigger gap between the two ratings tell us that some sale could be applied to the devices and the ratings might not be as high for these products if the selling price is the same or close to the original price. On the other hand, the highest ratings products have almost the same original and selling prices but still got the higher rating. There is a trend of increase of ratings as the price increases. We can assume that low price doesn't garantee good ratings becasue people are much happier to pay more to get the features and capacity up to their standards.

Share

make presentation

Act

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 ?

Observations:

- Overall steps per day for participants are more than average Americans.
- Participants use device to watch calories intake but not reduce calories intake.
- Device help reduce sedentary minutes but not increase active minutes.
- Should have a uniformed report method. Obvious weight loss or gain were not observed.
- Spend most of the time asleep so FitBit does help with increasing sleep.
- Fossil and Fire-boltt dominate among the high ratings.
- Black, blue, silver are the most popular colors.
- Fossil, Apple, and FitBit have overall highest ratings.
- Silicone, stainless steel, aluminum are top three favorite strap materials.
- Costumers are willing to pay more for good quality products with good ratings. Ratings and Price have a positive correlations.

Marketing Recommendations:

- Add features to enable using personally surveying tools like weekly emails to create a semi-customized workout/activity schedule in the Bellabeat app.
- Encourage users to use WIFI-scale report to report weight in order to get more accurate and uniformed data.
- Leaf trackers can increase steps by adding engaging games that involve meaningful prize. Can collaborate with charity to improve the brand image.
- Using Leaftracker, Bellabeat app, and Spring water bottle to create an environment that promotes healthy diet and help monitor water and calorie intake.
- Leaf trackers can use buzzes and reminders to remind the users to stand after long sedentary hours.
- Time can include features to guide and teaches users how to meditate before bed to improve sleep quality. Include different time and levels of meditation.
- Investigate into what features and characteristics that make Fossil, Apple, and FitBit such sucessful brands and learn from them.
- Make band straps with different materials available and make sure to include silicone, stainless steel, and aluminum straps.
- Advertise and create a product imgae of good quality fitness devices carefully crafted by professionals.