Bellabeat Case Study

By Dikchhya Palikhe

Introduction

This case study on a wellness company was prepared as a part of the Google Data Analytics Professional Certificate capstone. Bellabeat is a successful small company that manufactures health-focused products for women. In order to grow in the global smart device market, Urška Sršen, cofounder and Chief Creative Officer of Bellabeat wants to analyze the smart device data to help guide the marketing strategy of their company.

The analysis will be based on the following questions:

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?

By analyzing the smart device usage data, I was able to understand how consumers use non-Bellabeat smart devices and apply the findings in coming up with recommendations for Bellabeat marketing strategy.

Data

The dataset used in this project was downloaded from FitBit Fitness Tracker Data by Mobius on Kaggle. It contains personal tracker data such as minute-level output for physical activity, heart rate and sleep monitoring from thirty Fitbit users. The datasets were responses from a Amazon Mechanical Turk survey in 2016.

For the purpose of this study, I used three datasets:

1)Daily Activity has 940 observations and 15 variables and consists of steps, distance and calories tracked from the users.

2)Sleep Day has 413 observations and 5 variables and consists of total time spent sleeping and total time spent on the bed.

3)Weight Log has 67 observations and 8 variables and consists of weight, fat and BMI of the users.

Data Cleaning

Using Excel,I changed the data format of the variable SleepDay from Sleep Day data to (mm/dd/yy) to match that of Daily Activity data. Then, I imported the datasets to R for further data cleaning and analysis.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(dplyr)
```

```
daily_activity<- read.csv("DailyActivity.csv")
sleep_day<- read.csv("Sleepday1.csv")
weight_log<- read.csv("WeightLog.csv")
attach(daily_activity)
attach(sleep_day)
```

```
## The following object is masked from daily_activity:
##
##     Id
```

```
attach(weight_log)
```

```
## The following object is masked from sleep_day:
##
##     Id
##
## The following object is masked from daily_activity:
##
##     Id
```

```
#Removing missing values
daily_activity<- daily_activity%>%drop_na()
sleep_day<-sleep_day%>%drop_na()
weight_log<-weight_log%>%drop_na()
```

```
#Changing the minutes to hours and selecting only the required columns
daily_activity<-daily_activity %>% mutate(VeryActiveHours = VeryActiveMinutes/60) %>% mutate(FairlyActi

sleep_day<-mutate(sleep_day,TotalHoursAsleep = TotalMinutesAsleep / 60) %>% mutate(sleep_day, TotalHours
```

Analysis and Data Visualizations

```
#5 number summary
summary(daily_activity)
```

```
##        Id            ActivityDate         TotalSteps    TotalDistance
##  Min.   :1.504e+09   Length:940         Min.   :    0   Min.   : 0.000
##  1st Qu.:2.320e+09   Class :character   1st Qu.: 3790   1st Qu.: 2.620
##  Median :4.445e+09   Mode  :character   Median : 7406   Median : 5.245
##  Mean   :4.855e+09                      Mean   : 7638   Mean   : 5.490
##  3rd Qu.:6.962e+09                      3rd Qu.:10727   3rd Qu.: 7.713
##  Max.   :8.878e+09                      Max.   :36019   Max.   :28.030
##  TrackerDistance  VeryActiveHours   FairlyActiveHours LightlyActiveHours
##  Min.   : 0.000   Min.   :0.00000   Min.   :0.0000    Min.   :0.000
##  1st Qu.: 2.620   1st Qu.:0.00000   1st Qu.:0.0000    1st Qu.:2.117
##  Median : 5.245   Median :0.06667   Median :0.1000    Median :3.317
##  Mean   : 5.475   Mean   :0.35275   Mean   :0.2261    Mean   :3.214
##  3rd Qu.: 7.710   3rd Qu.:0.53333   3rd Qu.:0.3167    3rd Qu.:4.400
##  Max.   :28.030   Max.   :3.50000   Max.   :2.3833    Max.   :8.633
```

```
##  SedentaryHours     Calories
##  Min.   : 0.00   Min.   :   0
##  1st Qu.:12.16   1st Qu.:1828
##  Median :17.62   Median :2134
##  Mean   :16.52   Mean   :2304
##  3rd Qu.:20.49   3rd Qu.:2793
##  Max.   :24.00   Max.   :4900
```

```
summary(sleep_day)
```

```
##        Id              SleepDay         TotalSleepRecords TotalHoursAsleep
##  Min.   :1.504e+09   Length:413         Min.   :1.000     Min.   : 0.9667
##  1st Qu.:3.977e+09   Class :character   1st Qu.:1.000     1st Qu.: 6.0167
##  Median :4.703e+09   Mode  :character   Median :1.000     Median : 7.2167
##  Mean   :5.001e+09                      Mean   :1.119     Mean   : 6.9911
##  3rd Qu.:6.962e+09                      3rd Qu.:1.000     3rd Qu.: 8.1667
##  Max.   :8.792e+09                      Max.   :3.000     Max.   :13.2667
##  TotalHoursinBed
##  Min.   : 1.017
##  1st Qu.: 6.717
##  Median : 7.717
##  Mean   : 7.644
##  3rd Qu.: 8.767
##  Max.   :16.017
```

```
summary(weight_log)
```

```
##        Id                Date              WeightKg      WeightPounds
##  Min.   :1.504e+09   Length:2           Min.   :52.60   Min.   :116.0
##  1st Qu.:2.208e+09   Class :character   1st Qu.:57.55   1st Qu.:126.9
##  Median :2.912e+09   Mode  :character   Median :62.50   Median :137.8
##  Mean   :2.912e+09                      Mean   :62.50   Mean   :137.8
##  3rd Qu.:3.616e+09                      3rd Qu.:67.45   3rd Qu.:148.7
##  Max.   :4.320e+09                      Max.   :72.40   Max.   :159.6
##       Fat              BMI        IsManualReport     LogId
##  Min.   :22.00   Min.   :22.65   Mode:logical   Min.   :1.46e+12
##  1st Qu.:22.75   1st Qu.:23.85   TRUE:2         1st Qu.:1.46e+12
##  Median :23.50   Median :25.05                  Median :1.46e+12
##  Mean   :23.50   Mean   :25.05                  Mean   :1.46e+12
##  3rd Qu.:24.25   3rd Qu.:26.25                  3rd Qu.:1.46e+12
##  Max.   :25.00   Max.   :27.45                  Max.   :1.46e+12
```

```
length(unique(daily_activity$Id))
```

```
## [1] 33
```

```
length(unique(sleep_day$Id))
```

```
## [1] 24
```

```
length(unique(weight_log$Id))
```

```
## [1] 2
```

From the summary,I found out that there is a difference in between means of TotalDistance(5.490) and TrackerDistance(5.475) from the daily_activity dataset.SedentaryHours has a mean of about 16.52 hours while VeryActiveHours has a mean of just 0.35275.Similarly, from the sleep_day dataset, average TotalHoursAsleep is 6.9911 hours while the average TotalHoursinBed is 7.644 hours. This difference in hours suggest that

people must have difficulty in falling asleep or they must use their electronic device in bed.

Similarly, I found out that the dataset daily_activity has 33 distinct Ids, sleep_day has 24 Ids and weight_log has only 2 Ids. Therefore, I decided to use only the first two datasets for the rest of my analysis.

In order to merge daily_activity and sleep_day, I renamed their date column to a common name for convenience.

```
#Renaming columns
daily_activity<-daily_activity %>% rename(Date = ActivityDate)
sleep_day<-sleep_day %>% rename(Date = SleepDay)
```

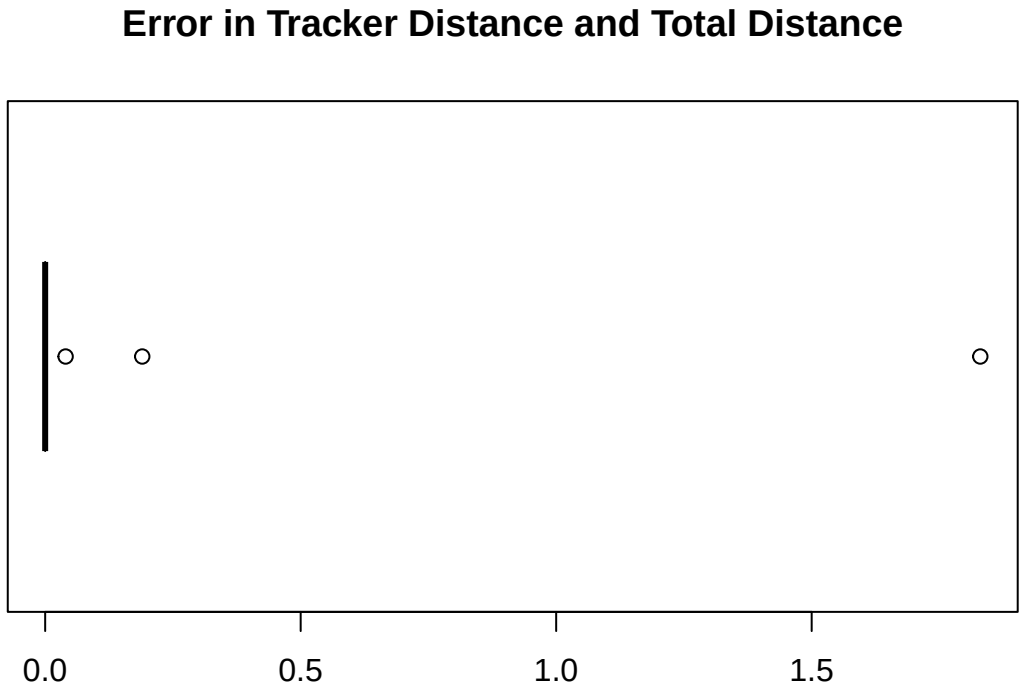
```
#Merging daily_activity and sleep_day using right join
activity_sleep<-daily_activity %>% right_join(sleep_day, by=c("Id","Date"))
```

Based on the 5 number summary, there is a difference between mean Total Distance and mean Tracker Distance. So, I wanted to find out if the error is negligible.

```
#Finding the difference(error) between Total Distance and Tracker Distance
activity_sleep<-activity_sleep %>% mutate( Errorindistance=(TotalDistance-TrackerDistance))
mean(activity_sleep$Errorindistance)
```

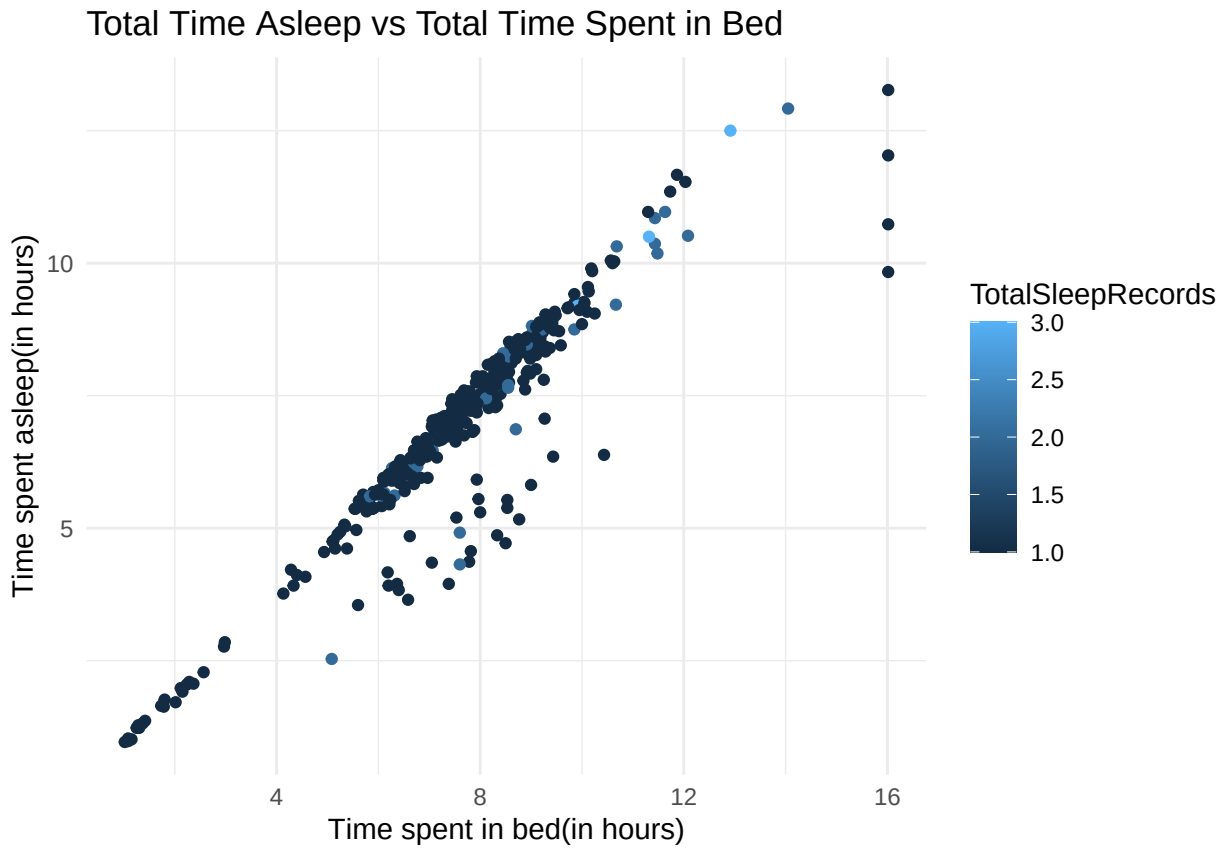
```
## [1] 0.004987894
```

```
#Plotting error in distance
boxplot(activity_sleep$Errorindistance, horizontal = TRUE, main="Error in Tracker Distance and Total Di
```



The average error in the distance is 0.01435106. So, we can accept the error in actual distance and distance tracked by the devices.

```
ggplot(activity_sleep,aes(x=TotalHoursinBed, y=TotalHoursAsleep,color=TotalSleepRecords))+geom_point()+
```

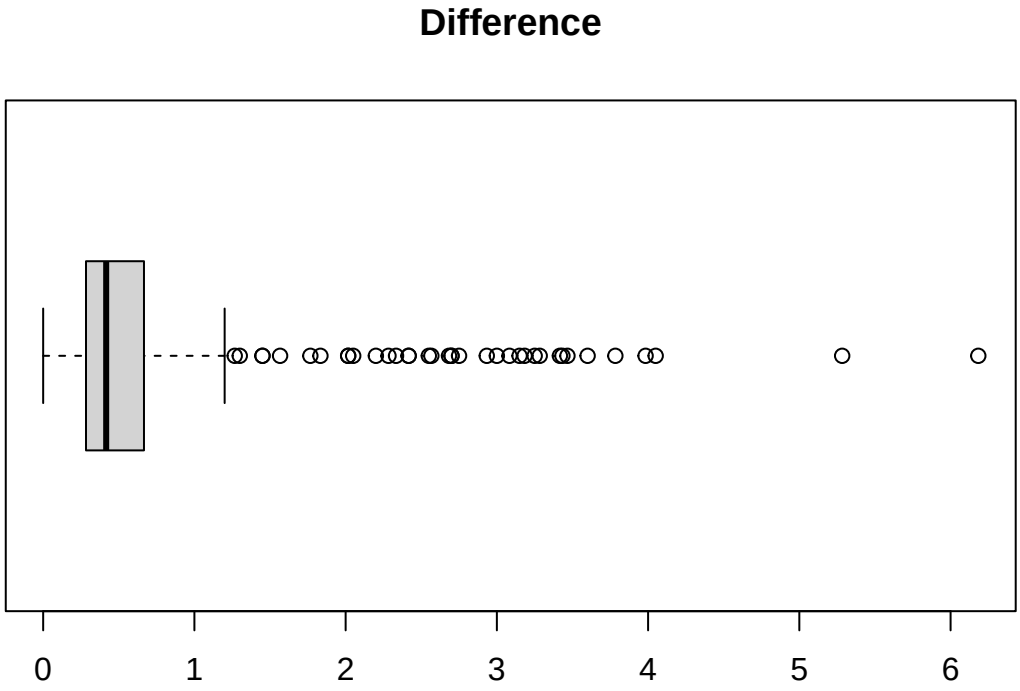


From the scatterplot, there is a positive correlation between time spent in bed and time spent asleep as estimated. Then, I looked at the difference between these times.

```
# Difference in sleeping hours and hours in bed
activity_sleep<-activity_sleep %>% mutate( DiffinBedHours=(TotalHoursinBed-TotalHoursAsleep))
mean(activity_sleep$DiffinBedHours)*60
```

```
## [1] 39.17191
```

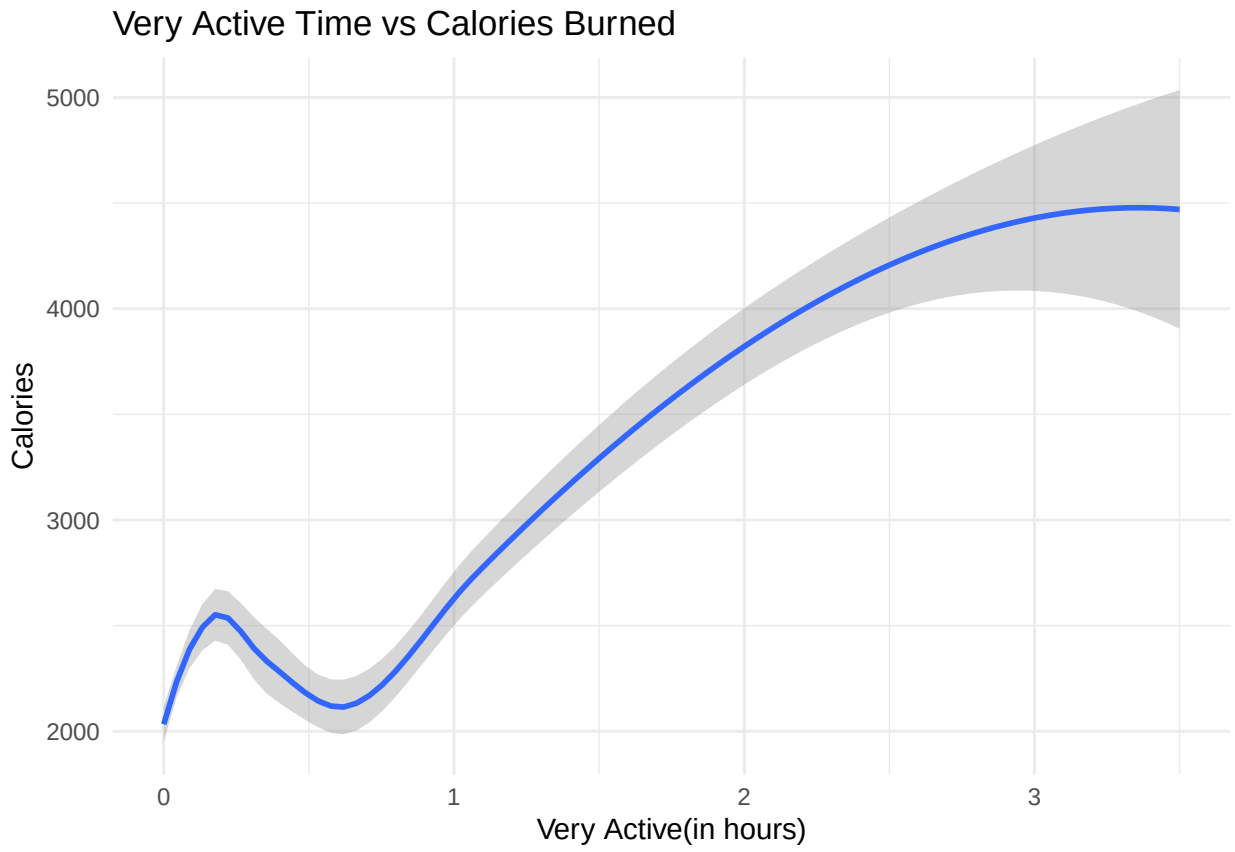
```
boxplot(activity_sleep$DiffinBedHours, horizontal = TRUE, main="Difference")
```



The average difference is about 39 minutes.

```
ggplot(activity_sleep,aes(x=VeryActiveHours, y=Calories))+geom_smooth()+labs(title="Very Active Time vs
```

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



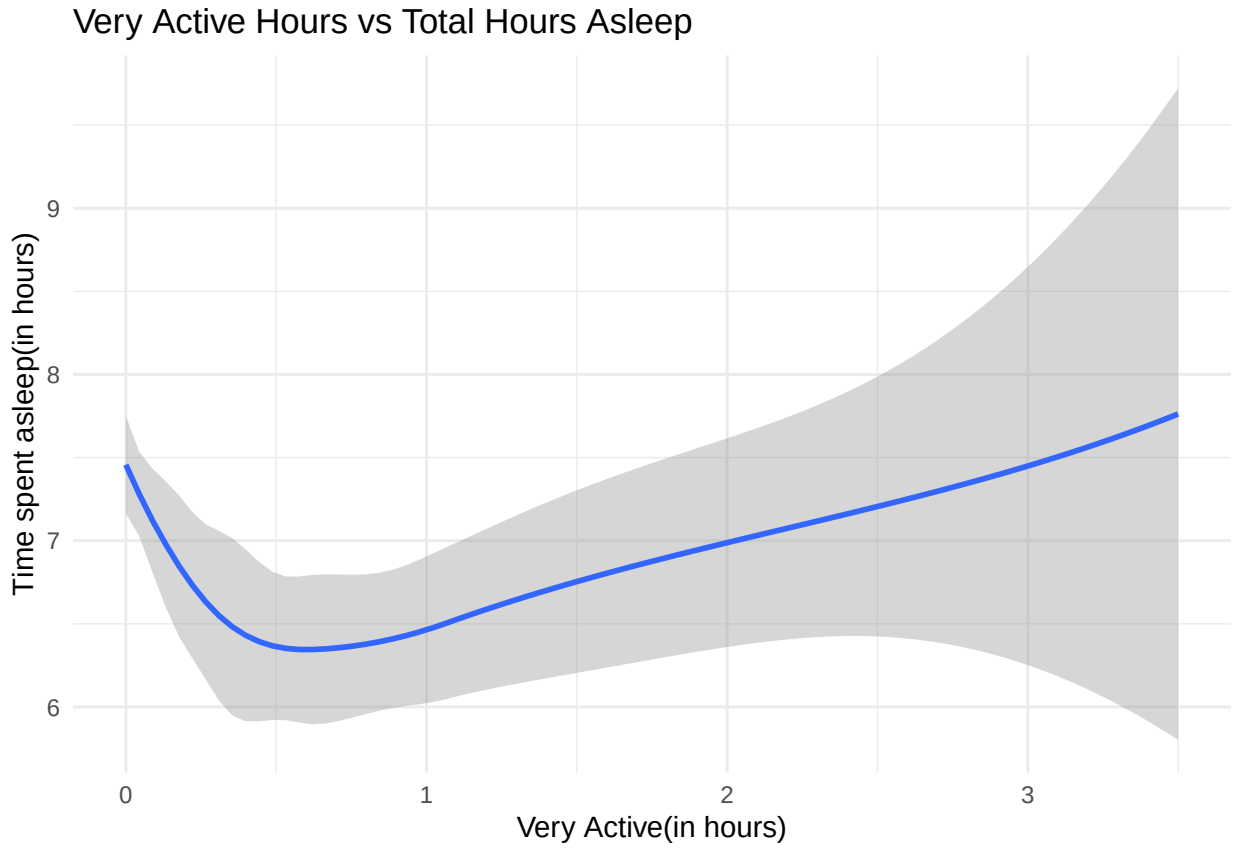
Next, I wanted to look at the relationship between active hours and calories burned. As estimated, there is a positive correlation-more active hours led to more calories burned.

```
ggplot(activity_sleep,mapping=aes(x=TotalSteps, y=Calories, color=TotalDistance))+geom_point()+labs(tit
```

Likewise, I found similar results between steps and calories burned.

```
ggplot(activity_sleep,aes(x=VeryActiveHours, y=TotalHoursAsleep))+geom_smooth()+labs(title="Very Active
```

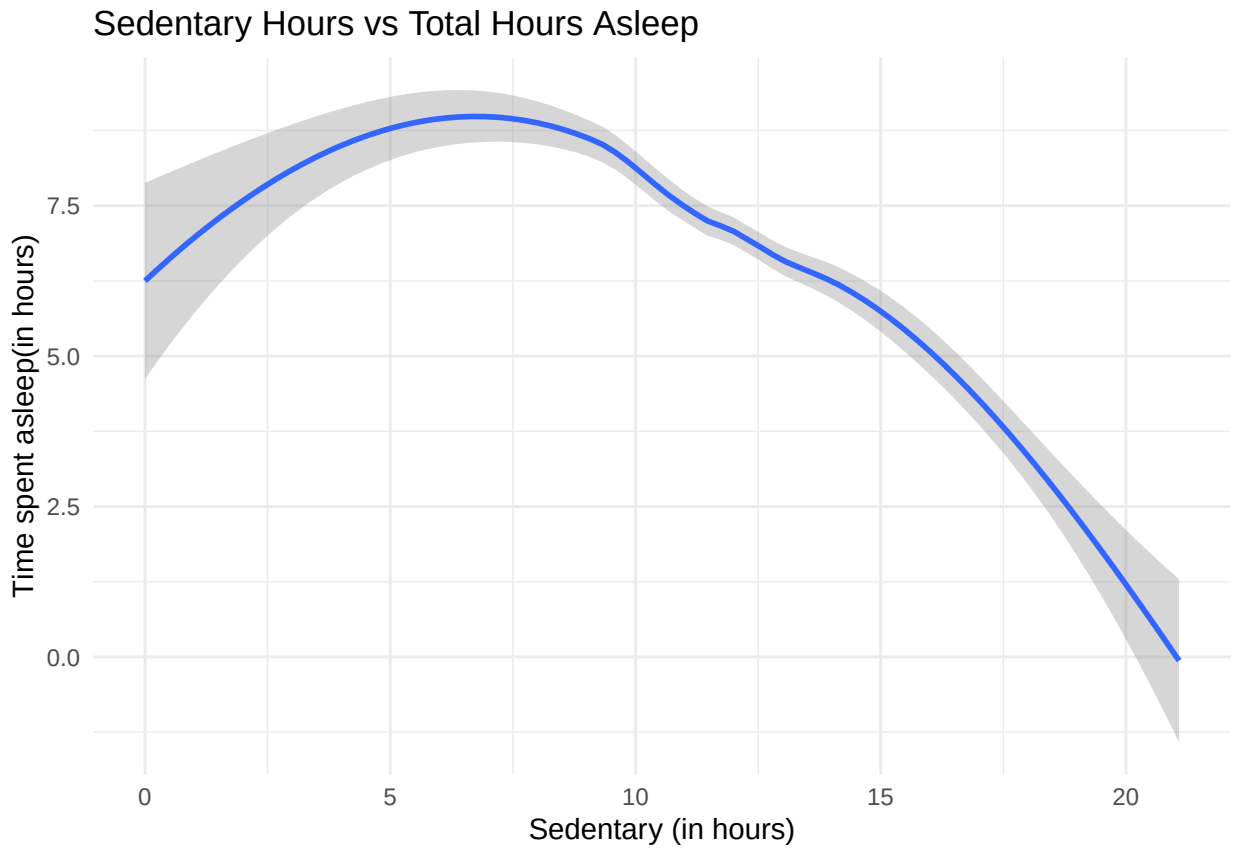
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



In addition to the calories, I wanted to check how active hours affected the users sleep. From the graph of "Very Active Hours vs Total Hours Asleep", we can say that there is also a positive correlation between the two variables. When users spent more time being active, they slept more and got the recommended hours of sleep..

```
ggplot(activity_sleep,aes(x=SedentaryHours, y=TotalHoursAsleep))+geom_smooth()+labs(title="Sedentary Hou
```

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



In the same way, looking at the graph of sedentary hours and total hours asleep, there is a negative correlation. The more hours the users spent inactive, the less sleep they got. The two graphs might suggest that being more active helps them to sleep better.

```
write.csv(activity_sleep, file = "activity_sleep.csv")
```

Recommendations

Bellabeat makes health-focused products such as Bellabeat app, Time, Leaf and Spring for women. The marketing team of Bellabeat could focus on women who are having a hard time in improving their lifestyle and becoming healthy. There can be a number of factors that is stopping these women from enjoying a healthy lifestyle that they want. Some of the factors could be poor time management and lack of progress results. Some women may have a lot in their plate and might not be able to track their time or remember exercising/sleeping on time. Similarly, others might lose interest in their initiative and find it as a waste of time due to the lack of progress data. Here Bellabeat's products could help these women to put themselves back in track ,and analyze and review their goals and progress reports. Time and Leaf will track the women's activity, sleep and stress and the users can view their goals and progress in the Bellabeat app. Similarly, users can also schedule time blocks on the app for exercise and sleep so that they will get notifications to remind them of their healthy lifestyle choice.In addition to this, based on their goals or their previous data, the app can also recommend the amount of time they should be active in order to get the recommended amount of sleep or vice versa. Besides, the app could also calculate the amount of sleep and exercise needed to burn certain amount of calories. Besides,with the stress and sleep tracked, the app can also potentially help women with sleep problems.

Clustering

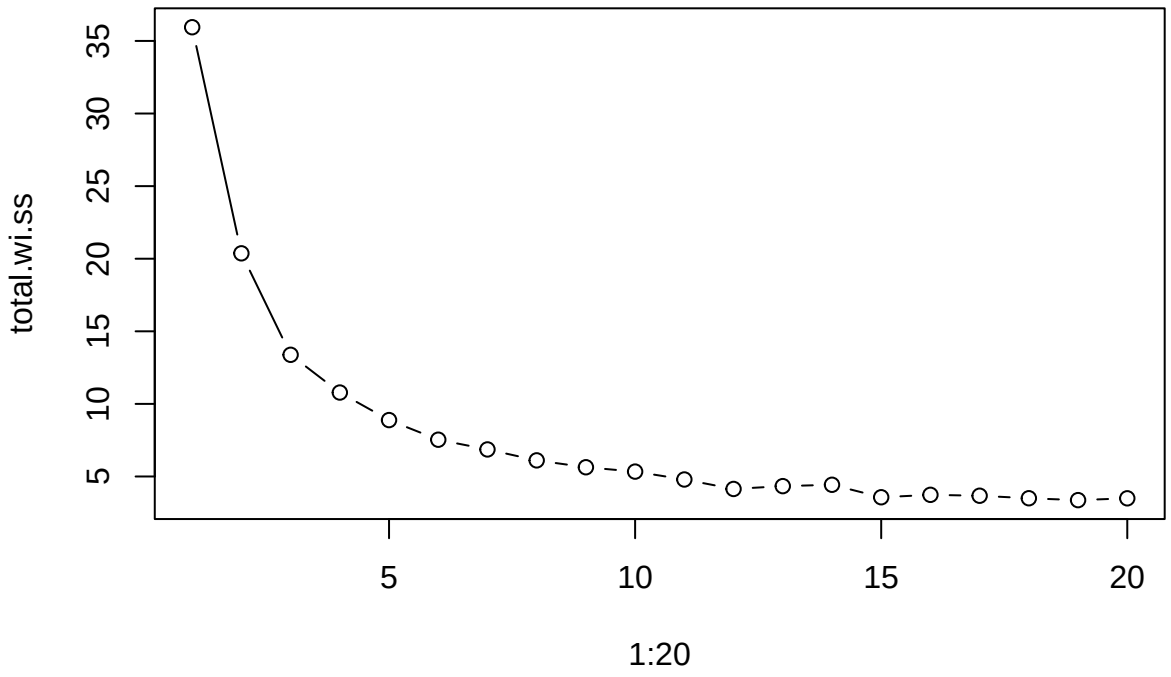
In order to divide the similar observations into groups, first I selected the columns that I wanted to work with. I created two similar dataframes(activity_sleep and activity_sleep_normed) so that I can normalise

the data (in activity_sleep_normed) and still have the original data.

```
activity_sleep<-activity_sleep%>%select(TrackerDistance, VeryActiveHours, Calories )
activity_sleep_normed<-activity_sleep%>%select(TrackerDistance, VeryActiveHours, Calories )
```

```
#normalising
for(i in 1:ncol(activity_sleep_normed)){
  min<-min(activity_sleep_normed[,i])
  max<-max(activity_sleep_normed[,i])
  for (j in 1:nrow(activity_sleep_normed)){
  activity_sleep_normed[j,i]<-(activity_sleep_normed[j,i]-min)/(max-min)
  }
}
```

```
#finding optimal number of clusters
total.wi.ss<-c()
for(i in 1:20){
  total.wi.ss[i]<-kmeans(activity_sleep_normed,centers=i)$tot.withinss
}
plot(x=1:20,y=total.wi.ss,type="b")
```



From the plot, I found that 5 is the optimal number of clusters for this dataset as the total within-cluster sum of squares starts decreasing in an almost linear fashion.

```
#using kmeans for 5
set.seed(1234)
activity_sleep.kmeans<-kmeans(activity_sleep_normed,centers=5)
activity_sleep.kmeans
```

```
## K-means clustering with 5 clusters of sizes 23, 112, 127, 96, 55
##
## Cluster means:
##   TrackerDistance VeryActiveHours  Calories
## 1       0.6238250     0.638302277 0.8138479
## 2       0.3439359     0.043792517 0.5247435
```

```
## 3       0.1494356     0.007461567 0.3309223
## 4       0.4596228     0.146676587 0.3608411
## 5       0.4717004     0.271341991 0.6590684
##
## Clustering vector:
##   [1] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 5 3 2 3 3 3 3 3 3 3 3
##  [38] 3 3 3 3 3 3 3 3 3 3 4 3 3 3 3 3 3 3 3 3 4 3 3 3 4 3 3 3 3 4 2 2 3 2 2 4 3
##  [75] 4 2 4 3 4 3 3 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 4 2 2
## [112] 2 5 5 2 2 2 2 3 3 3 2 2 3 3 3 3 2 2 2 2 2 2 3 2 2 2 2 3 2 2 2 3 2 2 2 5 2
## [149] 2 2 5 5 2 2 2 2 5 2 5 5 2 2 1 1 2 2 2 3 3 3 2 3 3 3 3 3 2 3 2 2 3 3 2 3 2
## [186] 2 3 3 3 3 2 3 3 2 3 4 2 3 3 5 2 2 2 2 2 5 2 2 2 2 5 5 2 2 2 2 2 2 2 2 2 2
## [223] 5 5 2 2 2 3 4 3 4 4 3 3 3 4 3 4 4 3 3 4 4 4 4 4 3 3 4 4 3 4 4 3 3 4 4 3 3
## [260] 5 2 1 1 1 1 1 1 5 1 5 2 1 5 2 5 5 5 1 1 2 5 2 5 5 3 4 2 3 2 4 5 2 2 2 2 3
## [297] 2 2 2 3 2 2 3 3 2 3 4 3 3 3 4 4 4 4 4 4 4 5 3 4 4 4 4 2 4 3 4 4 4 4 3 3 3
## [334] 4 4 3 3 3 3 5 2 2 2 5 5 2 5 3 5 5 5 5 5 2 5 5 5 5 2 2 5 5 3 5 1 1 5 1 1 5
## [371] 5 3 1 1 5 1 1 2 2 1 1 1 2 2 2 3 5 5 5 2 5 5 2 3 5 5 5 3 3 3 3 3 3 3 2 3 2
## [408] 2 2 3 3 3 3
##
## Within cluster sum of squares by cluster:
## [1] 1.353564 1.818263 1.719903 1.529480 2.380171
##  (between_SS / total_SS =  75.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
activity_sleep$Cluster<-activity_sleep.kmeans$cluster
```

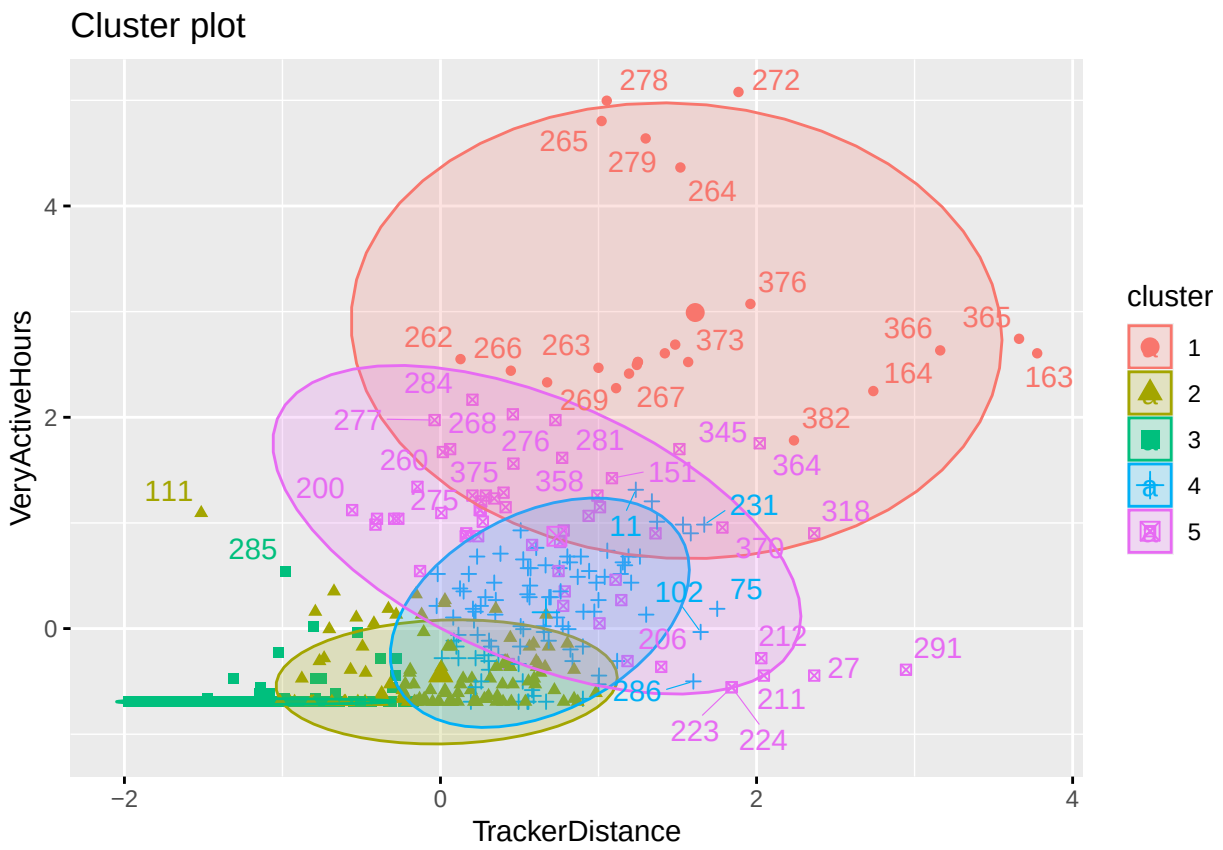
```
#renaming the clusters
activity_sleep$Cluster<-as.factor(activity_sleep$Cluster)
#levels(activity_sleep$Cluster)<-c("Most Active","Fairly Active","Active","Lightly Active","Least Activ
```

```
#plotting the clusters
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

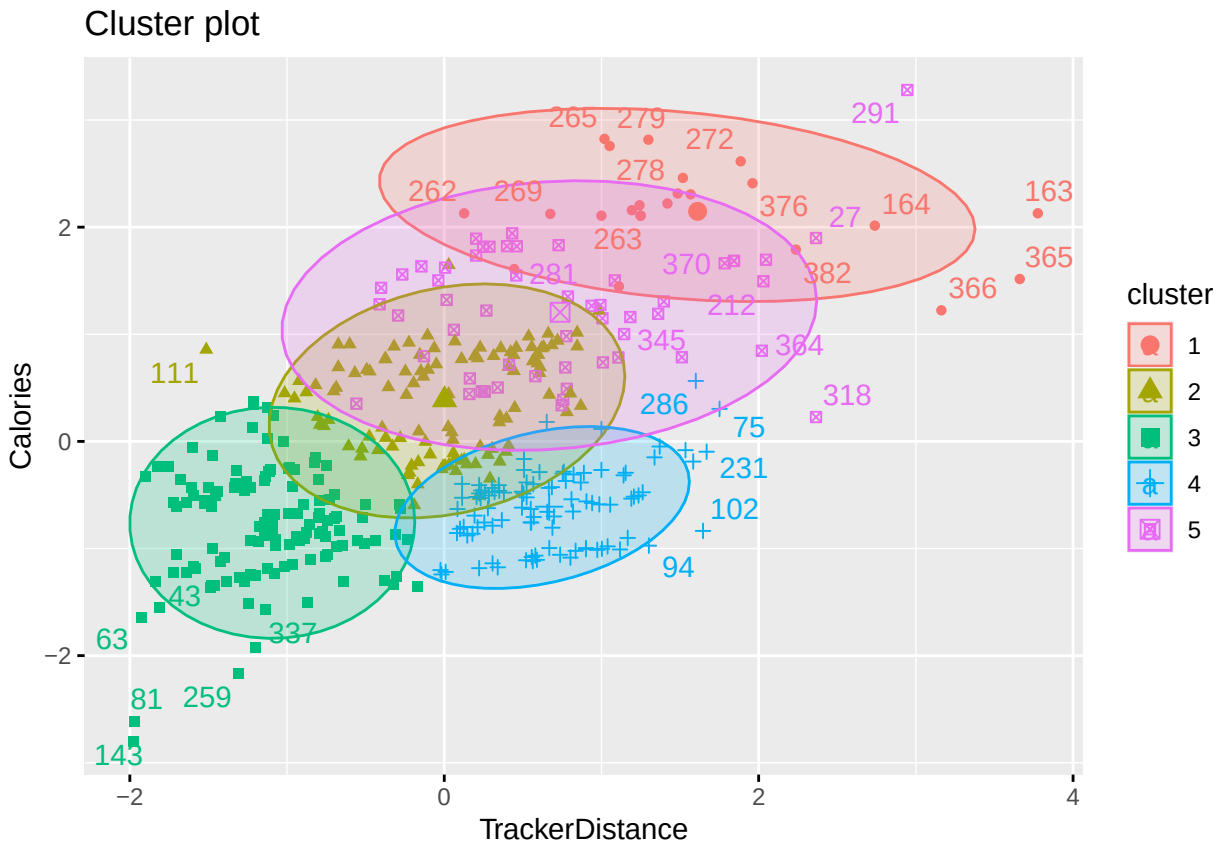
```
fviz_cluster(activity_sleep.kmeans,data=activity_sleep[,c(1,2)],repel=T,ellipse.type="t")
```

```
## Warning: ggrepel: 367 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```



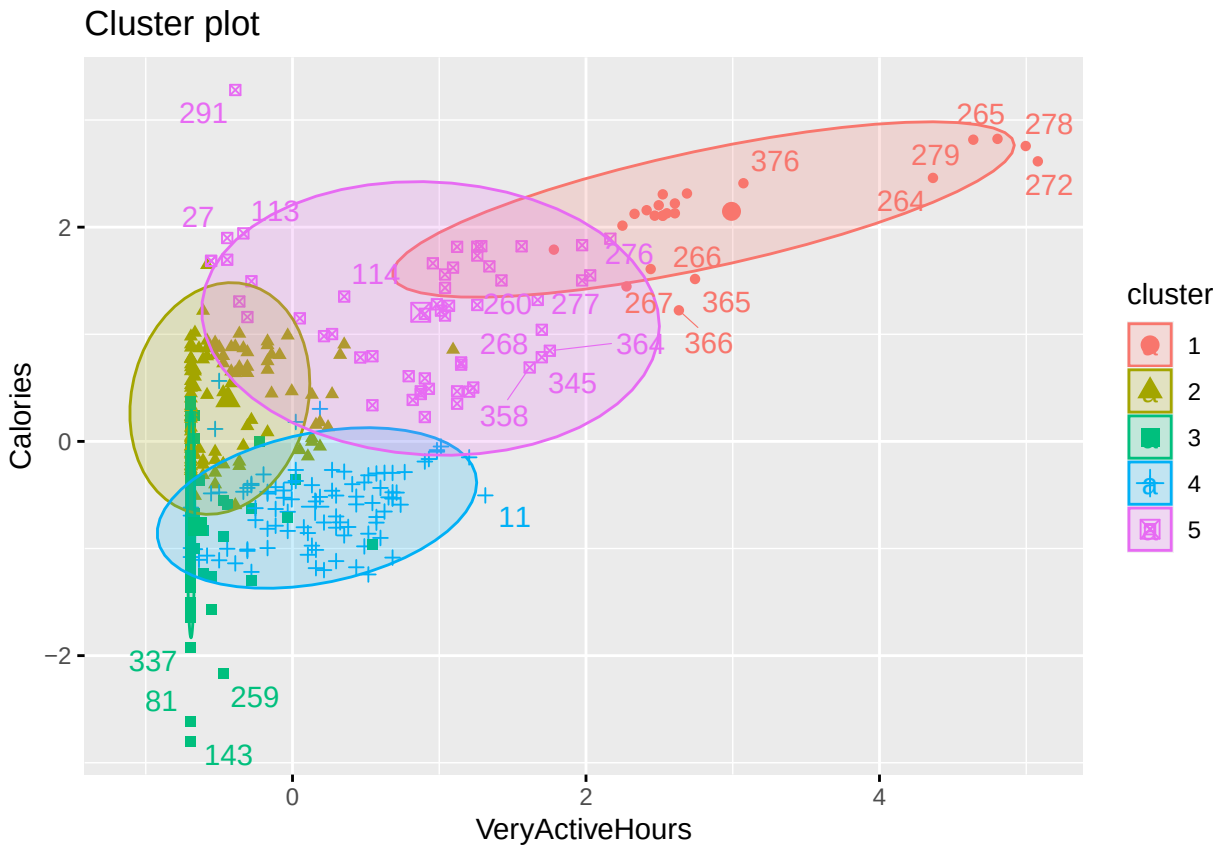

```
fviz_cluster(activity_sleep.kmeans,data=activity_sleep[,c(1,3)],repel=T,ellipse.type="t")
```

```
## Warning: ggrepel: 380 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

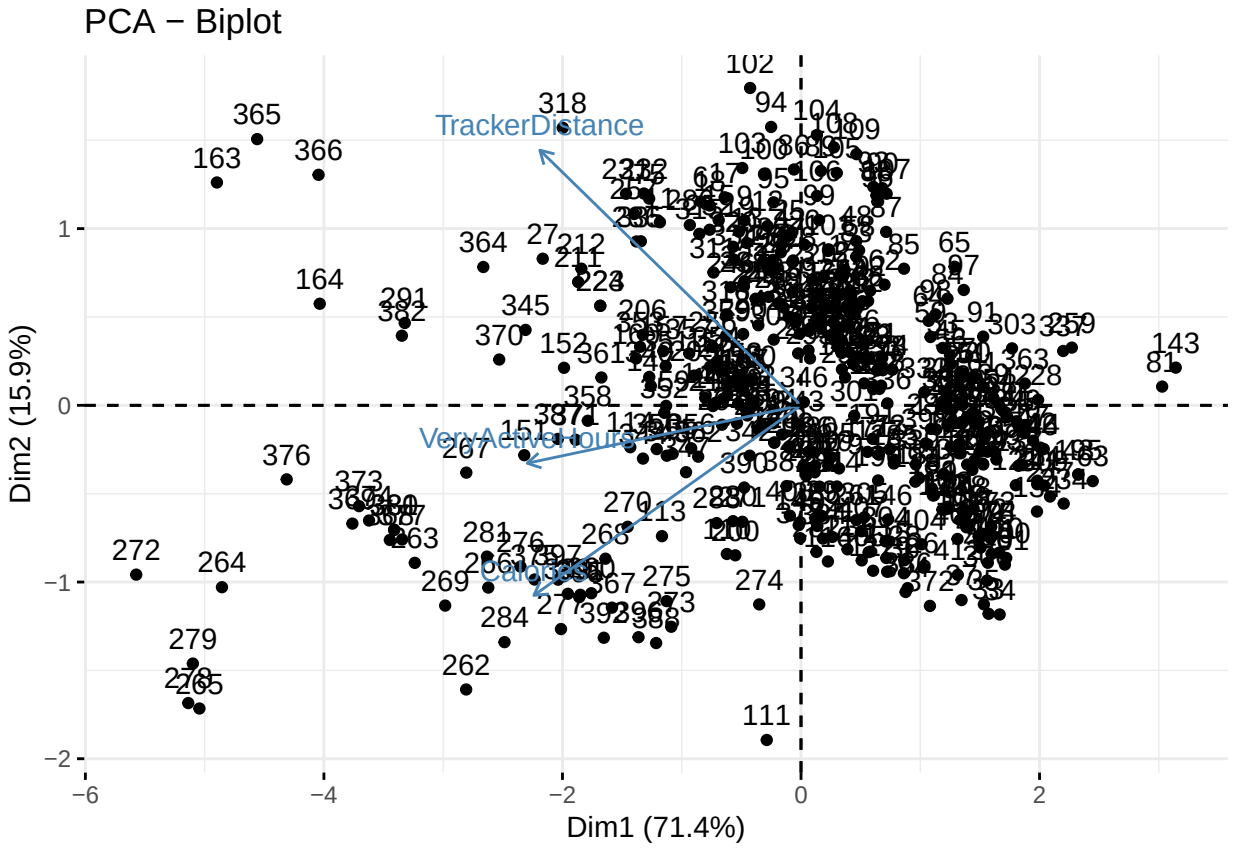



```
fviz_cluster(activity_sleep.kmeans,data=activity_sleep[,c(2,3)],repel=T,ellipse.type="t")
```

```
## Warning: ggrepel: 387 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```




```
activity.pca<-prcomp((activity_sleep)[,1:3],center=T,scale=T)
fviz_pca_biplot(activity.pca)
```




```
#summarisiing
activity_sleep.summary<-activity_sleep%>%group_by(Cluster)%>%summarise(average.TrackerDistance=mean(Tra
activity_sleep.summary
```

```
## # A tibble: 5 x 4
##   Cluster average.TrackerDistance average.VeryActiveHours average.Calories
##   <fct>                     <dbl>                   <dbl>            <dbl>
## 1 1                         10.9                   2.23             4036.
## 2 2                          6.04                  0.153            2693.
## 3 3                          2.63                  0.0261           1793.
## 4 4                          8.07                  0.513            1932.
## 5 5                          8.28                  0.950            3317.
```

Cluster 1 has average tracker distance as 10.945652, average very active hours as 2.23405797 and average calories as 4035.696. Cluster 2 has average tracker distance as 6.039196, average very active hours as 0.15327381 and average calories as 2693.384. Cluster 3 has average tracker distance as 2.629606, average very active hours as 0.02611549 and average calories as 1793.472. Cluster 4 has average tracker distance as 8.067187 , average very active hours as 0.51336806 and average calories as 1932.385. Cluster 5 has average tracker distance as 8.278909, average very active hours as 0.94969697 and average calories as 3317.055.