

# Analiza comportamentului utilizatorilor ce folosesc produsele Bellabeat

Nicolae Oana-Sorina

Neagu Andreea-Maria

Necula Mihaela-Diana

## What is the purpose/problem?

Bellabeat vrea să găsească noi oportunități pentru a-și dezvolta afacerea. Din acest motiv, vom analiza informații despre utilizatorii actuali care utilizează produsele oferite de companie (dispozitive inteligente). Misiunea este de a găsi tendințe în date și de a face recomandări utile companiei.

## Why Bellabeat?

Bellabeat este un producător de înaltă tehnologie de produse axate pe sănătate pentru femei. Scopul său este să devină un jucător important pe piața globală a dispozitivelor inteligente.

## What was done before?

În urma analizelor realizate pe această companie, s-a ajuns la următoarele concluzii: Bellabeat ar putea recomanda un număr minim de pași pe care utilizatorii să-i facă pentru a-i încuraja să-și atingă obiectivele. Datele arată că, pentru a dormi mai bine, cel mai bun tip de exercițiu este ușor până la moderat (mai puțin de 10.000 de pași). Studiile au arătat că o rutină înainte de culcare poate ajuta la obținerea unui somn mai bun. Bellabeat ar putea trimite un memento să înceapă acea rutină la o anumită oră și să recomande activități pentru relaxare și îmbunătățirea somnului. Bellabeat ar putea recomanda, de asemenea, câteva articole despre sănătate și fitness în cadrul aplicației Bellabeat, ca o modalitate excelentă de a motiva utilizatorii.

## What is your solution? + Results

Datele au fost preluate de pe site-ul Kaggle pentru a găsi o soluție în vederea dezvoltării afacerii Bellabeat și anume lărgirea oportunităților pentru a viza mai mulți utilizatori.

Pentru o analiză cât mai exactă, datele au fost curățate prin eliminarea variabilelor care nu aduceau informații noi, iar pentru un format cât mai adecvat am schimbat structura datei.

```
daily_activity <- daily_activity %>%
  rename(Date = ActivityDate) %>%
  mutate(Date = as.Date(Date, format = "%m/%d/%y"))
final_df <- daily_activity %>% select(-c(TrackerDistance))
```

În prima parte, s-a realizat o descriere a setului de date și o analiză a acestora din punct de vedere calitativ (statistici descriptive, matricea de corelație, studierea dependențelor dintre variabile). Apoi, s-au construit hărți cu auto-organizare SOM pentru clusterizare și clasificare în vederea structurării informației într-un număr adecvat de clustere pentru o mai bună vizualizare a datelor și realizarea unei predicții asupra clasificării clienților în funcție de activitatea fizică executată zilnic.

```
summary(daily_activity[,3:7])
```

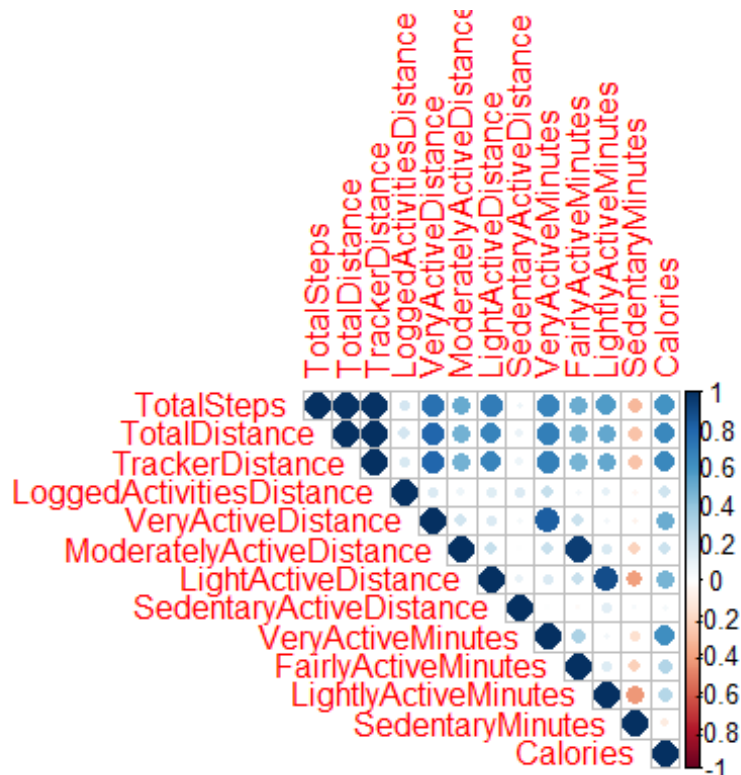
TotalSteps	TotalDistance	
Min. : 0	Min. : 0.000	
1st Qu.: 3790	1st Qu.: 2.620	
Median : 7406	Median : 5.245	
Mean : 7638	Mean : 5.490	
3rd Qu.: 10727	3rd Qu.: 7.713	
Max. : 36019	Max. : 28.030	

LightlyActiveMinutes	SedentaryMinutes	Calories
Min. : 0.0	Min. : 0.0	Min. : 0
1st Qu.: 127.0	1st Qu.: 729.8	1st Qu.: 1828
Median : 199.0	Median : 1057.5	Median : 2134
Mean : 192.8	Mean : 991.2	Mean : 2304
3rd Qu.: 264.0	3rd Qu.: 1229.5	3rd Qu.: 2793
Max. : 518.0	Max. : 1440.0	Max. : 4900

În medie, utilizatorii parcurg un număr de 7406 pași, dintre care 25% se deplasează mai puțin de 2.6 km (Q1), iar 75% mai puțin de 7.7 km (Q3). Pe eșantionul preluat, clienții Bellabeat ard în medie 2304 calorii, 25% dintre aceștia ard mai puțin de 1828 kcal, iar 75 % mai mult de 2793 kcal.

```
matr_corelatie<-cor(daily_activity[,3:15])
corrplot(matr_corelatie, method=c("circle"), type=c("upper"))
```



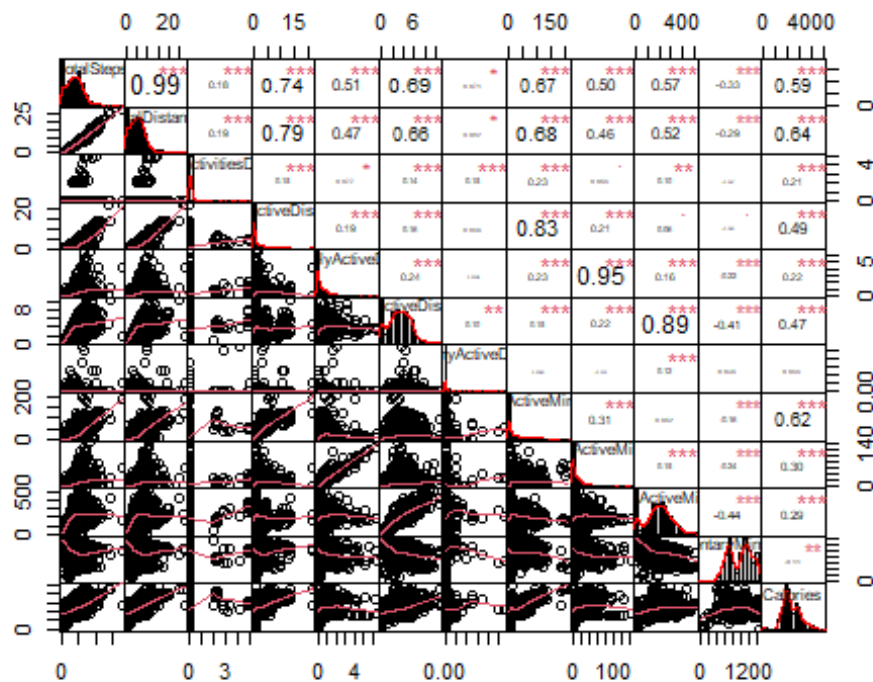
În matricea de corelație se identifică legături puternice și directe atât între numărul total de pași, distanța totală parcursă și distanța monitorizată cât și între distanța activă moderată și timpul în care utilizatorul a fost destul de activ. Se observă și legături de intensitate medie între variabilele TotalDistance, TotalSteps și VeryActiveDistance, LightActiveDistance.

```
par(mfrow=c(2,2))
boxplot(daily_activity$TotalSteps,main="Boxplot numar total de
pasi",col="aquamarine3")
boxplot(daily_activity$TotalDistance,main="Boxplot distante",col="pink")
boxplot(daily_activity$Calories,main="Boxplot calorii",col="red")
```

În toate cele 3 boxplot-uri realizate pe variabilele TotalSteps, TotalDistance și Calories se observă prezența outlierilor pe valorile mari. Outlierii pentru numărul total de pași sunt apropiați de 30000, iar pentru distanța totală de 28 km.

```
chart.Correlation(final_df[,3:14], histogram = TRUE, pch=19)
```

În figura de mai jos, se observă legături directe între majoritatea variabilelor, excepție făcând legătura dintre `LightlyActiveMinutes` și `SedentaryMinutes`, aceasta prezentând un coeficient de -0.44 .



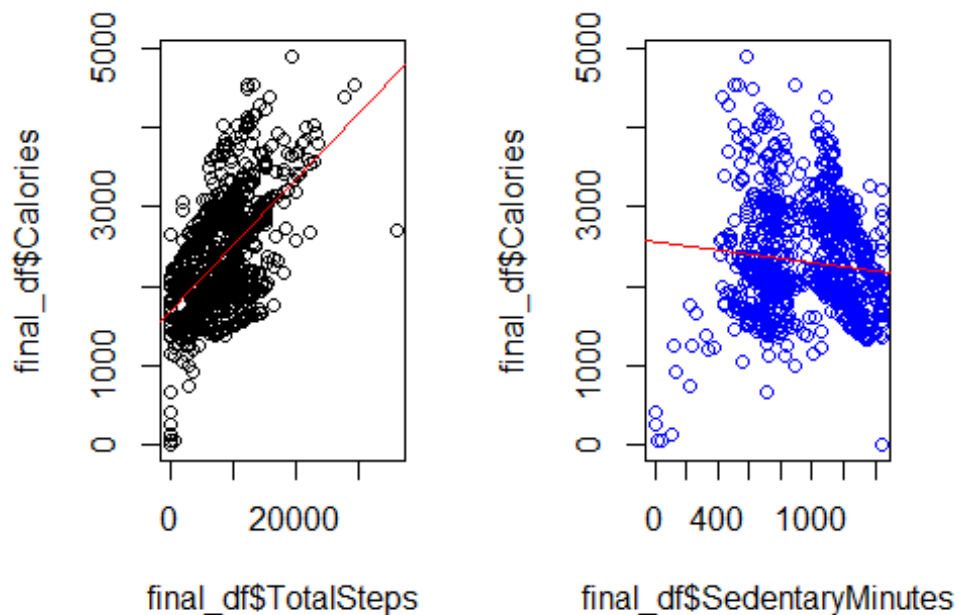
```

par(mfrow=c(1,2))
plot(final_df$TotalSteps, final_df$Calories, main="Calorii arse in functie de
numarul total de pasi")
abline(lm(final_df$Calories~final_df$TotalSteps), col="red")

plot(final_df$SedentaryMinutes, final_df$Calories, col="blue", main="Calorii
arse in functie de distanta totala parcursa")
abline(lm(final_df$Calories~final_df$SedentaryMinutes), col="red")

```

În exemplul de mai jos se observă legatura directă între numărul total de pași și cel de calorii arse, cele mai frecvente valori regăsindu-se între 1000 și 3000 kcal la un număr de maxim de 20000 pași parcurși. În ceea ce privește legatura dintre timpul în care utilizatorii sunt sedentari și kaloriile arse, aceasta este una indirectă.



```

daily_average <- daily_activity %>%
  group_by(Id) %>%
  summarise (mean_daily_steps = mean(TotalSteps), mean_daily_calories =
mean(Calories))

```

```
head(daily_average)
```

```
# A tibble: 6 × 3
      Id mean_daily_steps mean_daily_calories
  <dbl>         <dbl>         <dbl>
1 1503960366      12117.         1816.
2 1624580081       5744.         1483.
```

```
3 1644430081          7283.          2811.
4 1844505072          2580.          1573.
5 1927972279           916.          2173.
6 2022484408        11371.          2510.
```

```
user_type <- daily_average %>%
  mutate(user_type = case_when(
    mean_daily_steps < 5000 ~ "sedentary",
    mean_daily_steps >= 5000 & mean_daily_steps < 7499 ~ "lightly active",
    mean_daily_steps >= 7500 & mean_daily_steps < 9999 ~ "fairly active",
    mean_daily_steps >= 10000 ~ "very active"
  ))
```

```
View(user_type)
```

```
user_type_percent <- user_type %>%
  group_by(user_type) %>%
  summarise(total = n()) %>%
  mutate(totals = sum(total)) %>%
  group_by(user_type) %>%
  summarise(total_percent = total / totals) %>%
  mutate(labels = scales::percent(total_percent))
```

```
user_type_percent$user_type <- factor(user_type_percent$user_type , levels =
c("very active", "fairly active", "lightly active", "sedentary"))
```

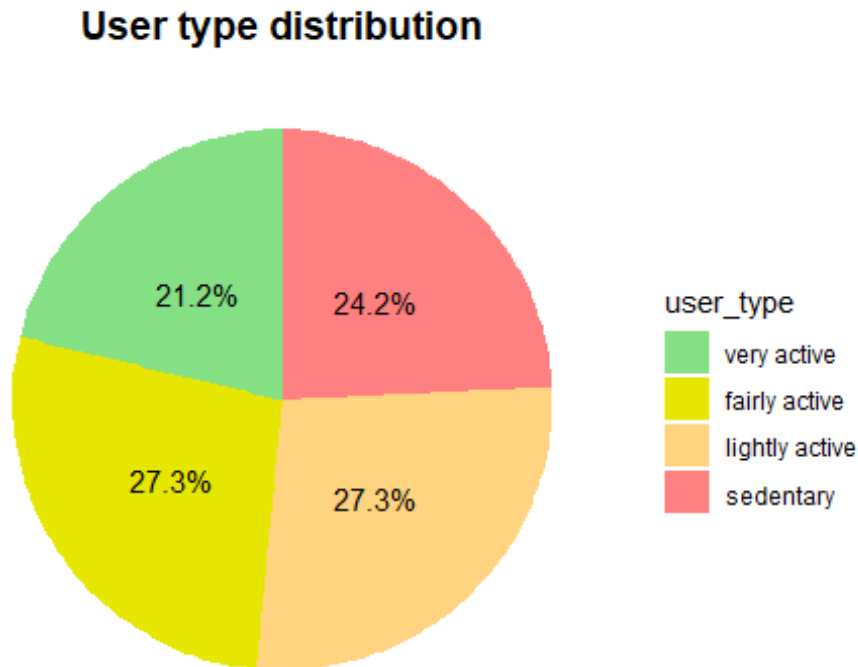
```
head(user_type_percent)
```

```
# A tibble: 4 × 3
  user_type      total_percent labels
  <fct>          <dbl> <chr>
1 fairly active    0.273 27.3%
2 lightly active    0.273 27.3%
3 sedentary        0.242 24.2%
4 very active      0.212 21.2%
```

```
user_type_percent %>%
  ggplot(aes(x="",y=total_percent, fill=user_type)) +
  geom_bar(stat = "identity", width = 1)+
  coord_polar("y", start=0)+
  theme_minimal()+
  theme(axis.title.x= element_blank(),
        axis.title.y = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.ticks = element_blank(),
        axis.text.x = element_blank(),
        plot.title = element_text(hjust = 0.5, size=14, face = "bold")) +
  scale_fill_manual(values = c("#85e085", "#e6e600", "#ffd480", "#ff8080")) +
```

```
geom_text(aes(label = labels),
          position = position_stack(vjust = 0.5))+
labs(title="User type distribution")
```

În continuare, se realizează o distribuție a tipurilor de utilizatori, aceasta împărțindu-se în patru categorii: foarte activi (21.2%), destul de activi (27.3%), ușor activi (27.3%) și sedentari (24.2%).



```
weekday_steps <- daily_activity %>%
  mutate(weekday = weekdays(Date))

weekday_steps$weekday <- ordered(weekday_steps$weekday, levels=c("Monday",
"Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday", "Sunday"))

weekday_steps <- weekday_steps %>%
  group_by(weekday) %>%
  summarize (daily_steps = mean(TotalSteps), daily_distance =
mean(TotalDistance))

head(weekday_steps)

# A tibble: 6 × 3
  weekday   daily_steps daily_distance
```

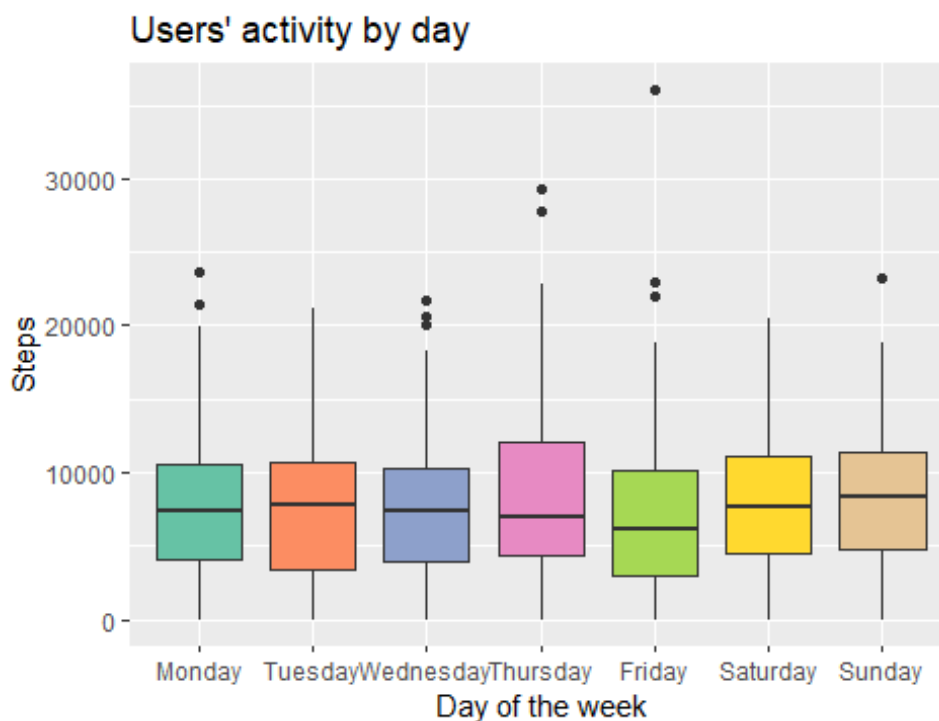
	<ord>	<dbl>	<dbl>
1	Monday	7559.	5.49
2	Tuesday	7406.	5.31
3	Wednesday	7448.	5.31
4	Thursday	8153.	5.85
5	Friday	6933.	5.03
6	Saturday	7781.	5.55

```

final_df %>%
  mutate(weekdays = weekdays(Date)) %>%
  select(weekdays, TotalSteps) %>%
  mutate(weekdays = factor(weekdays, levels = c('Monday', 'Tuesday',
'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'))) %>%
  drop_na() %>%
  ggplot(aes(weekdays, TotalSteps, fill = weekdays)) +
  geom_boxplot() +
  scale_fill_brewer(palette="Set2") +
  theme(legend.position="none") +
  labs(
    title = "Users' activity by day",
    x = "Day of the week",
    y = "Steps"
  )

```

În această reprezentare se pot identifica diferențele de activitate în decursul unei săptămâni. Se remarcă faptul că majoritatea utilizatorilor parcurg mai mulți pași în zilele de weekend.





```

daily_activity <- within(daily_activity, {
  Calories_byday <- NA # need to initialize variable
  Calories_byday[daily_activity$Calories<=500] <- "Sedentary"
  Calories_byday[500<daily_activity$Calories & daily_activity$Calories<=2000]
<- "Lightly active"
  Calories_byday[2000<daily_activity$Calories & daily_activity$Calories<=3000]
<- "Fairly active"
  Calories_byday[daily_activity$Calories>3000] <- "Very active"
} )

View(daily_activity)
train.obs<-sample(nrow(daily_activity), 752)
train.obs

```

Am împărțit de date în set de antrenare (752 de observații) și de testare (288 de observații).

```

[1] 332 124 636 616 885 161 935 36 92 376 87 940 205 233 793 237 419 611
[19] 88 689 899 513 609 277 549 160 587 583 402 841 200 694 308 875 83 828
[37] 249 90 642 338 97 767 102 621 309 562 920 753 272 889 531 901 520 346
[55] 1 431 800 130 105 350 290 627 741 214 103 571 7 228 318 866 720 217
[73] 744 84 738 224 602 342 423 24 188 868 667 666 435 287 348 256 795 117
[91] 505 26 564 18 305 743 618 748 40 926 559 724 613 442 381 148 456 852
[109] 162 289 405 264 466 838 560 137 13 202 749 790 502 140 349 501 101 536

date_som<-daily_activity[,3:15]
View(date_som)
train.set<-scale(date_som[train.obs,],[-13])
train.set

```

Se normalizeaza datele setului de antrenare, urmand sa cream setul de testare. Se va normaliza pe baza mediei si abaterii standard a setului regasit in outputul de mai sus.

	TotalSteps	TotalDistance	TrackerDistance	LoggedActivitiesDistance
332	-0.076548793	-0.1039593481	-0.1014325701	-0.1722739
124	-1.344476414	-1.2576740246	-1.2588758693	-0.1722739
636	-0.262407715	-0.1847446649	-0.1824789721	-0.1722739
616	-0.116639828	-0.0433703305	-0.0406477383	-0.1722739
	VeryActiveDistance	ModeratelyActiveDistance	LightActiveDistance	
332	-0.560097032	-0.152883334	0.633163399	
124	-0.560097032	-0.634978646	-1.379585985	
636	-0.560097032	-0.634978646	0.681663339	

616	-0.560097032	-0.634978646	0.953263467
	SedentaryActiveDistance	VeryActiveMinutes	FairlyActiveMinutes
332	-0.2134698	-0.64083129	-0.16965558
124	-0.2134698	-0.64083129	-0.66958112
636	-0.2134698	-0.64083129	-0.66958112
616	-0.2134698	-0.64083129	-0.66958112
	LightlyActiveMinutes	SedentaryMinutes	
332	-0.135226805	-0.917040370	
124	-1.255554350	-0.850922340	
636	1.031395267	0.484661865	
616	1.411010385	0.160683518	

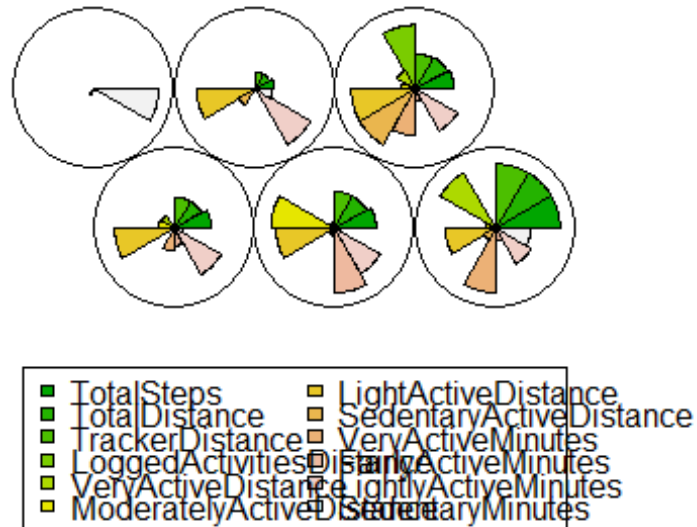
```
center = attr(train.set, "scaled:center")
scale = attr(train.set, "scaled:scale")
test.set<-scale(date_som[-train.obs, ],-13], center = attr(train.set,
"scaled:center"),scale
= attr(train.set, "scaled:scale"))
test.set
```

	TotalSteps	TotalDistance	TrackerDistance	LoggedActivitiesDistance
6	0.412328288	0.259574397	0.26327606	-0.1722739
10	0.434514589	0.224230873	0.22781831	-0.1722739
15	1.200525811	0.842743075	0.84832944	-0.1722739
23	0.683818546	0.428718582	0.43296689	-0.1722739
	VeryActiveDistance	ModeratelyActiveDistance	LightActiveDistance	
6	0.62072099	0.260341214	-0.390186265	
10	-0.06407943	-0.233232547	0.647713497	
15	0.30238129	0.421039708	1.079363312	
23	0.35050244	0.363647394	0.245163530	
	SedentaryActiveDistance	VeryActiveMinutes	FairlyActiveMinutes	
6	-0.2134698	0.48670981	0.33026996	
10	-0.2134698	-0.07706074	-0.26964069	
15	-0.2134698	0.27900487	0.48024762	
23	-0.2134698	0.42736554	0.43025507	
	LightlyActiveMinutes	SedentaryMinutes		
6	-0.246333669	-1.49557313		
10	0.188834882	-0.50710858		
15	0.818440444	-0.52363809		
23	0.114763639	0.62020383		

```
somexemplu <- som(train.set, grid = somgrid(3, 2, "hexagonal"))
windows()
plot(somexemplu)
```

Se construiește harta SOM cu  $3 \times 2 = 6$  clustere:

## Codes plot



Observam ca fiecare cluster este descris de unele variabile predominante, corespunzatoare triunghiurilor colorate cu cele mai mari marimi. Al doilea rând, reprezentat de ultimele trei clustere, este caracterizat de numarul total de pași (TotalSteps), distanța totală (TotalDistance), dar și monitorizarea distanței (TrackerDistance). Pe primul rând, sunt predominante LightActiveDistance, SedentaryActiveDistance și LightlyActiveMinutes. Primul cluster este reprezentat de variabila care înregistrează timpul în care utilizatorii sunt sedentari.

```
somprediction<-predict(somexemplu, newdata=test.set, trainX=train.set)
somprediction
```

Pentru realizarea clusterelor ce contin etichetele mentionate am ales sa afisam mai putine observatii pentru o reprezentare mai clara a clasificarii acestora.

```
$predictions
$predictions[[1]]
  TotalSteps TotalDistance TrackerDistance LoggedActivitiesDistance
6    0.7424576    0.6396239    0.6445538    -0.1722739
10   -0.1504590   -0.1746169   -0.1723184    -0.1722739
15    0.7424576    0.6396239    0.6445538    -0.1722739
23    0.7424576    0.6396239    0.6445538    -0.1722739
```

	VeryActiveDistance	ModeratelyActiveDistance	LightActiveDistance
6	0.3729460	0.4897896	0.5346319
10	-0.4313210	-0.4011050	0.4124885
15	0.3729460	0.4897896	0.5346319
23	0.3729460	0.4897896	0.5346319

	SedentaryActiveDistance	VeryActiveMinutes	FairlyActiveMinutes
6	-0.18408230	0.5309057	0.5144531
10	0.13550692	-0.4807640	-0.4077061
15	-0.18408230	0.5309057	0.5144531
23	-0.18408230	0.5309057	0.5144531

	LightlyActiveMinutes	SedentaryMinutes
6	0.36153783	-0.3080585
10	0.64234040	-0.2487705
15	0.36153783	-0.3080585
23	0.36153783	-0.3080585

Executam predictia clasificării tipului de utilizatori din setul de testare:

```
$unit.classif
[1] 1 5 1 1 1 1 1 5 5 5 1 2 4 5 4 1 5 5 4 4 4 4 4 4 1 1 1 1 1 1 1 4 5 5 5
5
[38] 5 5 5 5 5 4 4 5 5 4 2 1 5 4 5 5 5 5 5 5 5 5 1 2 2 5 1 1 2 4 4 4 4 1 5
4
[75] 5 4 5 5 5 1 1 1 3 3 5 5 5 5 1 5 5 5 5 5 5 5 4 5 1 1 1 1 4 5 1 1 5 5 1 1
5
[112] 1 1 1 5 1 4 1 5 5 4 5 4 5 5 5 5 4 4 4 4 1 4 4 5 1 1 5 1 1 5 6 6 1 6 6 6
1
[149] 4 3 3 1 4 3 4 2 4 1 4 4 1 6 3 5 6 5 5 5 4 2 4 4 2 1 4 4 4 5 5 4 4 4 3 3
1
[186] 5 1 3
```

```
$unit.predictions
$unit.predictions[[1]]
      TotalSteps TotalDistance TrackerDistance LoggedActivitiesDistance
[1,]  0.7424576    0.6396239    0.6445538      -0.1722739
[2,]  0.9323319    0.8921294    0.8978754      -0.1722739
[3,]  2.1257734    2.4165671    2.4272398      -0.1722739
[4,] -1.1170724   -1.0518759   -1.0524127     -0.1722739
[5,] -0.1504590   -0.1746169   -0.1723184     -0.1722739
[6,]  0.7718398    0.8509226    0.7704236       5.0097243
      VeryActiveDistance ModeratelyActiveDistance LightActiveDistance
```

```

[1,] 0.3729460 0.4897896 0.5346319
[2,] -0.1200665 3.3616777 0.4773573
[3,] 3.3442619 0.1802863 0.2102187
[4,] -0.5016290 -0.5447907 -1.1628141
[5,] -0.4313210 -0.4011050 0.4124885
[6,] 0.7326581 0.3769624 0.5415955
SedentaryActiveDistance VeryActiveMinutes FairlyActiveMinutes
[1,] -0.18408230 0.5309057 0.5144531
[2,] -0.12622563 -0.1966765 3.2938911
[3,] 0.03707759 2.6702746 0.3097602
[4,] -0.11376217 -0.5440436 -0.5553124
[5,] 0.13550692 -0.4807640 -0.4077061
[6,] 1.07076459 1.5442247 0.2302849
LightlyActiveMinutes SedentaryMinutes
[1,] 0.36153783 -0.3080585
[2,] 0.32251033 -0.4773555
[3,] -0.03432848 0.0573529
[4,] -1.17588367 0.7017179
[5,] 0.64234040 -0.2487705
[6,] 0.19290880 -0.4842317

```

```

$whatmap
[1] 1

```

Se observă că a doua observație aparține clusterului 5, iar următoarele primului cluster. Fiecare observație va fi grupată într-una din cele 6 clase.

```

somprediction$unit.classif

[1] 1 5 1 1 1 1 1 5 5 5 1 2 4 5 4 1 5 5 4 4 4 4 4 4 1 1 1 1 1 1 1 4 5 5 5
5
[38] 5 5 5 5 5 4 4 5 5 4 2 1 5 4 5 5 5 5 5 5 5 1 2 2 5 1 1 2 4 4 4 4 4 1 5
4
[75] 5 4 5 5 5 1 1 1 3 3 5 5 5 5 1 5 5 5 5 5 5 5 4 5 1 1 1 1 4 5 1 1 5 5 1 1
5
[112] 1 1 1 5 1 4 1 5 5 4 5 4 5 5 5 5 4 4 4 4 1 4 4 5 1 1 5 1 1 5 6 6 1 6 6 6
1
[149] 4 3 3 1 4 3 4 2 4 1 4 4 1 6 3 5 6 5 5 5 4 2 4 4 2 1 4 4 4 5 5 4 4 4 3 3
1
[186] 5 1 3

data.frame(somprediction$unit.classif)

```

```

      somprediction.unit.classif
1          1
2          5
3          1
4          1
5          1
6          1
7          1
8          5
9          5
10         5

id_unic<-paste(daily_activity$Id,daily_activity$Date,sep="-")
df<-data.frame(id_unic, daily_activity[,3:15])
View(df)
data <- data.frame(df[, -1],
                    row.names = df[,1])
View(data)
type<-rownames(data)
type

 [1] "1503960366-2020-04-12" "1503960366-2020-04-13" "1503960366-2020-04-14"
 [4] "1503960366-2020-04-15" "1503960366-2020-04-16" "1503960366-2020-04-17"
 [7] "1503960366-2020-04-18" "1503960366-2020-04-19" "1503960366-2020-04-20"
[10] "1503960366-2020-04-21" "1503960366-2020-04-22" "1503960366-2020-04-23"
[13] "1503960366-2020-04-24" "1503960366-2020-04-25" "1503960366-2020-04-26"
[16] "1503960366-2020-04-27" "1503960366-2020-04-28" "1503960366-2020-04-29"

View(daily_activity)

data<-data.frame(data, daily_activity[,16])
View(data)
testset<-type[-train.obs]
testset

 [1] "1503960366-2020-04-17" "1503960366-2020-04-21" "1503960366-2020-04-26"
 [4] "1503960366-2020-05-04" "1503960366-2020-05-06" "1503960366-2020-05-09"
 [7] "1503960366-2020-05-10" "1624580081-2020-04-12" "1624580081-2020-04-18"
[10] "1644430081-2020-04-15" "1644430081-2020-04-16" "1644430081-2020-04-19"
[13] "1644430081-2020-04-22" "1644430081-2020-04-23" "1644430081-2020-04-24"
[16] "1644430081-2020-04-26" "1644430081-2020-05-01" "1844505072-2020-04-13"

data.frame(testset,somprediction$unit.classif)

      testset somprediction.unit.classif
1 1503960366-2020-04-17                1
2 1503960366-2020-04-21                5
3 1503960366-2020-04-26                1

```

4	1503960366-2020-05-04	1
5	1503960366-2020-05-06	1
6	1503960366-2020-05-09	1
7	1503960366-2020-05-10	1
8	1624580081-2020-04-12	5
9	1624580081-2020-04-18	5
10	1644430081-2020-04-15	5

Utilizatorii 3-6 aparțin primului cluster, iar clusterul 5 cuprinde utilizatorii cu id-urile 1503960366(2020-04-21), 1624580081(2020-04-12, 2020-04-18), 1644430081(2020-04-15).

```
nb<-table(somprediction$unit.classif)
nb
```

Primul cluster contine 50 observatii, al doilea 8, cel de al cincilea are cele mai multe observatii (67), iar cel de al șaselea cele mai putine (7). Clustere 2 și 3 contin un numar apropiat de observatii (8 respectiv 9).

```
1 2 3 4 5 6
50 8 9 47 67 7

testset<-data[-train.obs,14]
testset
```

Primii utilizatori se încadrează în categoria “Lightly active”, următorii aparținând celei de “Fairly active”, fiind clasa predominantă. Cel mai rar întâlnit tip de utilizator este acela de “Very active”.

```
[1] "Lightly active" "Lightly active" "Lightly active" "Lightly active"
[5] "Lightly active" "Lightly active" "Lightly active" "Lightly active"
[9] "Lightly active" "Fairly active" "Very active" "Very active"
[13] "Fairly active" "Fairly active" "Fairly active" "Very active"
[17] "Fairly active" "Lightly active" "Lightly active" "Fairly active"
[21] "Fairly active" "Fairly active" "Fairly active" "Fairly active"
[25] "Fairly active" "Fairly active" "Fairly active" "Fairly active"
[29] "Fairly active" "Fairly active" "Fairly active" "Lightly active"
```

```
names(data)[14] <- 'calories_byday'

View(data)
daily_activ<-daily_activity[,3:16]
View(daily_activ)
tab<- table(data[-train.obs,]$calories_byday,somprediction$unit.classif)
tab
```

Din cei 188 de utilizatori extrași pentru predicție, cei mai mulți au fost încadrați în cel de al 5-lea cluster. Tipul cel mai des întâlnit de utilizator este reprezentat de cel destul de activ, iar cel mai rar întâlnit este cel sedentar.

	1	2	3	4	5	6
Fairly active	23	2	1	19	33	3
Lightly active	11	4	0	26	31	0
Sedentary	0	0	0	2	0	0
Very active	16	2	8	0	3	4

## Conclusions + Recomandations

În urma analizei realizate, Bellabeat ar putea să se axeze mai mult asupra unui anumit tip de utilizator, și anume cei din categoria “destul de activi” (“Fairly active”). Bellabeat ar putea recomanda acest nivel de exerciții pentru persoanele care doresc pur și simplu să mențină un stil de viață sănătos, adică nu fac sporturi de nivel înalt.

Pentru a crea un obicei de a face mișcare în fiecare zi, Bellabeat ar putea trimite o notificare la o anumită oră pentru ca utilizatorul să rămână consecvent pe tot parcursul săptămânii.