

Bellabeat

Bellabeat, a high-tech manufacturer of health-focused smart products including Bellabeat app, Leaf, Time, and Spring. In addition, Bellabeat also offers a subscription-based membership program for users giving them access to fully personalized guidance on nutrition, activity, sleep, health, and mindfulness based on their lifestyle and goals.

Prepare the data

The data set that have been used in this analysis is FitBit Fitness Tracker Data made by Mobius available on Kaggle. This data set is under CCO: Public Domain license meaning the creator has waived right to the work under the copyright law. The data set contains personal fitness tracker from thirty fitbit users. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It includes information about daily activity, steps, and heart rate that can be used to explore user's habits.

The following files from the dataset have been selected

- dailyActivity_merged.csv
- sleepDay_merged.csv
- weightLogInfo_merged.csv

Process the data

Google spreadsheet, Google BigQuery, and Google Data Studio will be used to process the data as the tool functionality fits the purpose.

Data Cleaning

sleepDay.csv and weight.csv are loaded into Google sheet for data cleaning. The fields "SleepDay" and "Date" were not correctly formatted. The following steps have been done:

1. The date column has been selected and formatted to "Date" using spreadsheet function.
2. Time in the column has been removed as time is irrelevant in this analysis.
3. AM/PM indicator is also removed.
4. And for the daily_activity I added column day using spreadsheet function Workday.

Data Integrity

The selected data has been loaded into Google BigQuery for analysis. The following queries have been run to check the number of unique Id in each table.

```
#Show total of the user daily_activity
SELECT COUNT(DISTINCT Id) AS Total_User
FROM `model-signifier-321912.Bellabeat.daily_activity`;

#Show total of the user sleepDay
SELECT COUNT(DISTINCT Id) AS Total_User
FROM `model-signifier-321912.Bellabeat.sleepDay`;

#Show total of the weight
SELECT COUNT(DISTINCT Id) AS Total_User
FROM `model-signifier-321912.Bellabeat.weight`;

#Show total of the dailyIntensities
SELECT COUNT(DISTINCT Id) AS Total_User
FROM `model-signifier-321912.Bellabeat.dailyIntensities`;
```

Results(DISTINCT Id in each table):

- 33
- 24
- 8
- 33

For the date range it from 12-04-2016 – 12-05-2016 and for the days it 31 days. For the query like this.

```
#Select the Date
SELECT MIN(ActivityDate) AS Star_Day, MAX(ActivityDate) AS End_Day
FROM `model-signifier-321912.Bellabeat.daily_activity`;

#Show the number of the days
SELECT COUNT(DISTINCT ActivityDate) AS Number_of_Days
FROM `model-signifier-321912.Bellabeat.daily_activity`;
```

Result:

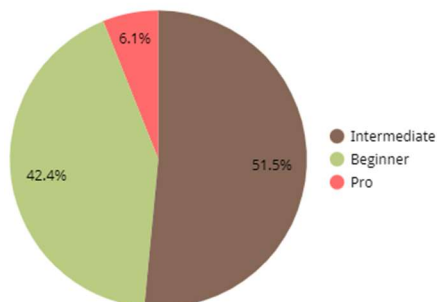
- 12-04-2016 – 12-05-2016
- 31

Analyze

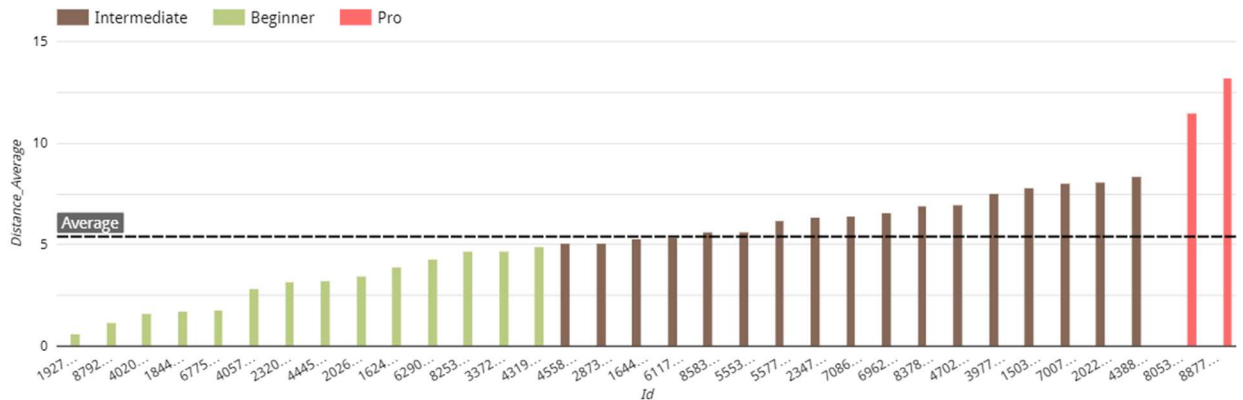
- Type of Runners

Now let us categorize runners according to their activity. For this purpose I will create three user types: *Beginner*, *Intermediate*, and *Pro*. For the *Beginners*, if a user runs on distance average less than 5 km it will fall in the *Beginner* group. If the distance average is between 5 km and 10 km, it will fall in the *Intermediate* group. Lastly, a runner considered to be *Professional* if the distance average is greater than 10 km.

```
#Classification
SELECT Id, ROUND(AVG(TotalDistance), 2) AS Distance_Average, COUNT(DISTINCT ActivityDate) AS Number_of_Days,
CASE WHEN ROUND(AVG(TotalDistance), 2) < 5 THEN 'Beginner'
WHEN ROUND(AVG(TotalDistance), 2) >= 5 AND ROUND(AVG(TotalDistance), 2) <= 10 THEN 'Intermediate'
ELSE 'Pro' END AS User_Type
FROM `model-signifier-321912.Bellabeat.daily_activity`
GROUP BY Id
ORDER BY User_Type;
```



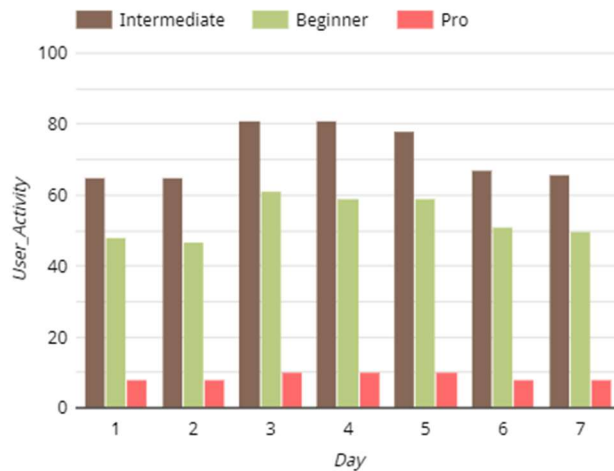
In order to graphically show the distribution of running distance, I also make the graph about the distance distribution.



- User activity per Day

Moving on to other variables, let's find out about the activity runners type by day. Using the previously added day column.

```
#User activity by Day
SELECT ac.Day, COUNT(*) AS User_Activity, a.User_Type
FROM (SELECT Id,
CASE WHEN ROUND(AVG(TotalDistance), 2) < 5 THEN 'Beginner'
WHEN ROUND(AVG(TotalDistance), 2) >= 5 AND ROUND(AVG(TotalDistance), 2) <= 10 THEN 'Intermediate'
ELSE 'Pro' END AS User_Type
FROM `model-signifier-321912.Bellabeat.daily_activity`
GROUP BY Id) AS a
INNER JOIN `model-signifier-321912.Bellabeat.daily_activity` AS ac
ON a.Id = ac.Id
GROUP BY Day, a.User_Type
ORDER BY Day, a.User_Type;
```

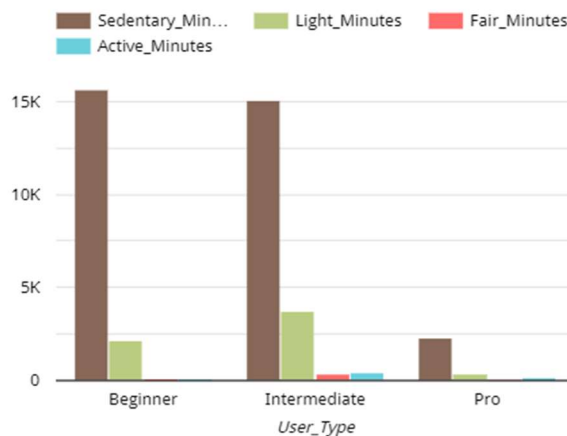


Note: 1 is Sunday and 7 is Saturday

- Intensity Minutes by User Type

Now take a closer look at the average activity intensity minutes by user type.

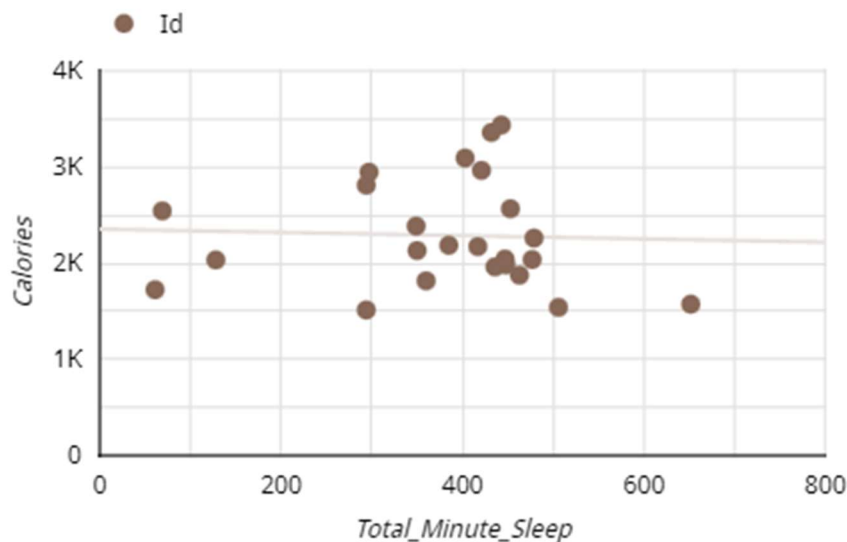
```
#Average Intesity Minutes
SELECT Id, ROUND(AVG(VeryActiveMinutes), 2) AS Active_Minutes,
ROUND(AVG(FairlyActiveMinutes), 2) AS Fair_Minutes,
ROUND(AVG(LightlyActiveMinutes), 2) AS Light_Minutes,
ROUND(AVG(SedentaryMinutes), 2) AS Sedentary_Minutes,
CASE WHEN ROUND(AVG(TotalDistance), 2) < 5 THEN 'Beginner'
WHEN ROUND(AVG(TotalDistance), 2) >= 5 AND ROUND(AVG(TotalDistance), 2) <= 10 THEN 'Intermediate'
ELSE 'Pro' END AS User_Type,
FROM `model-signifier-321912.Bellabeat.daily_activity`
GROUP BY Id
ORDER BY User_Type;
```



- Calories and Total Minute a Sleep Relation

We can study the relationship between calories burnt and total minute a sleep relation. Firstly, lets analyze the distribution of daily average hours of sleep.

```
#Calories and Total Minute Sleep
SELECT sleep.Id, ROUND(AVG(activity.Calories),2) AS Calories, ROUND(AVG(sleep.TotalMinutesAsleep)) AS Total_Minute_Sleep
FROM `model-signifier-321912.Bellabeat.daily_activity` AS activity
INNER JOIN `model-signifier-321912.Bellabeat.sleepDay` AS sleep
ON activity.Id = sleep.Id
GROUP BY sleep.Id
```

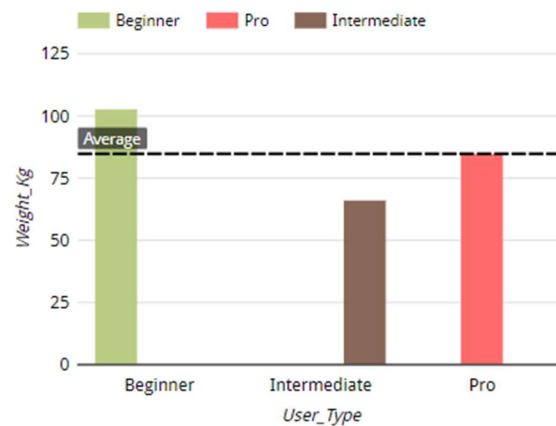


This visualization compares a person's calories burnt versus the minutes asleep. We can see that the more time a person spent in sleep, the lower the calories burnt. It is a negative relation which implies sleep negatively impact calories burnt.

- Average Weight

Now lets we find out the average weight of the user.

```
#Weight
SELECT DISTINCT a.id, ROUND(AVG(w.WeightKg),1) AS Weight_Kg,
CASE WHEN ROUND(AVG(a.TotalDistance), 2) < 5 THEN 'Beginner'
WHEN ROUND(AVG(a.TotalDistance), 2) >= 5 AND ROUND(AVG(a.TotalDistance), 2) <= 10 THEN 'Intermediate'
ELSE 'Pro' END AS User_Type
FROM `model-signifier-321912.Bellabeat.daily_activity` AS a
INNER JOIN `model-signifier-321912.Bellabeat.weight` AS w
ON a.Id = w.Id
GROUP BY a.Id;
```



From this table we only got 8 user type: Beginner 2, Intermediate 5, and Pro 1.

Conclusion

- In the categorization there are 14 beginners, 17 intermediate and 2 pro. This categorization is based on the average total distance traveled for 31 days.
- Most activities are carried out on Tuesdays based on user activities for all types of users.
- There is no negative relation between the average number of calories burned and the user's average sleep time.

Recommendation

Based on the analysis that I have done, then we have provided my final recommendations are:

- Make all important data such as sleep schedule, weight, height, and age is mandatory in order to sign up.
- Make a weekly dashboard as a result of user activities as historical performance, so that users can improve their activities.
- Create a notification for the user to increase their movement regularly to reduce their sedentary minutes.

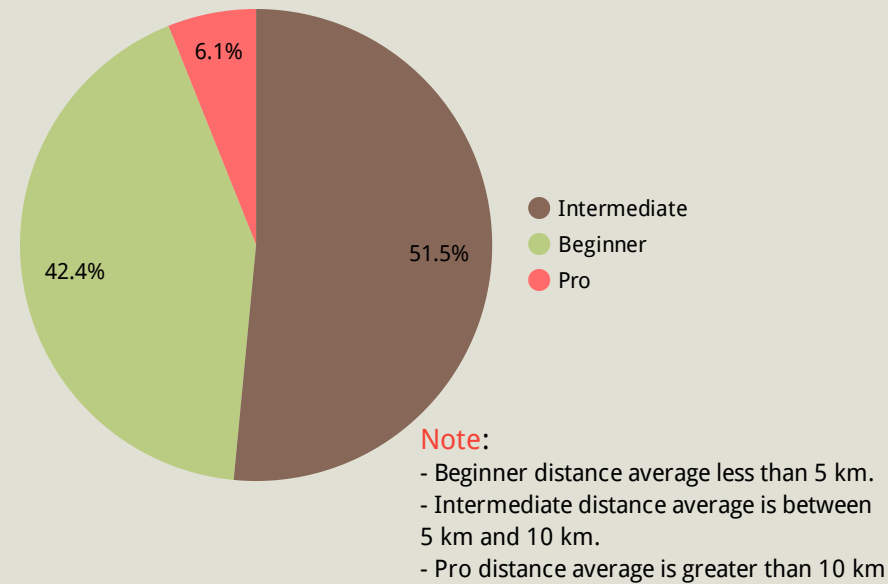
Bellabeat

Total_User
33

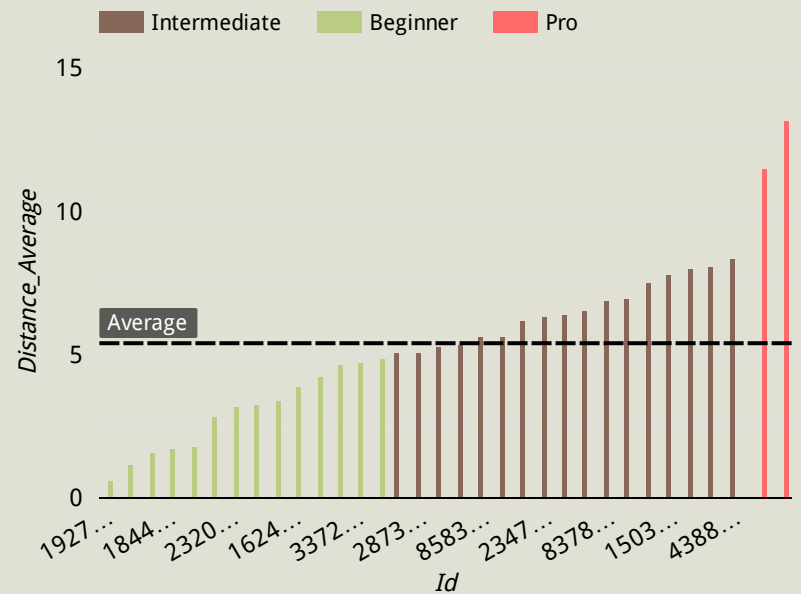
Number_of_Days
31

Bellabeat, a high-tech manufacturer of health-focused products for women, in this case we analyze data from smart device fitness data. In this data there are 33 different people without knowing their gender who use these smart devices for 31 days.

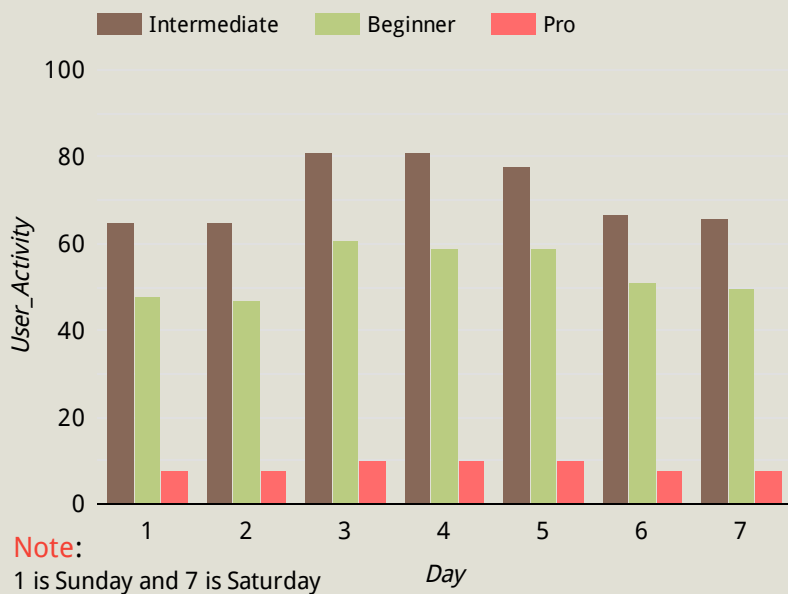
User Type



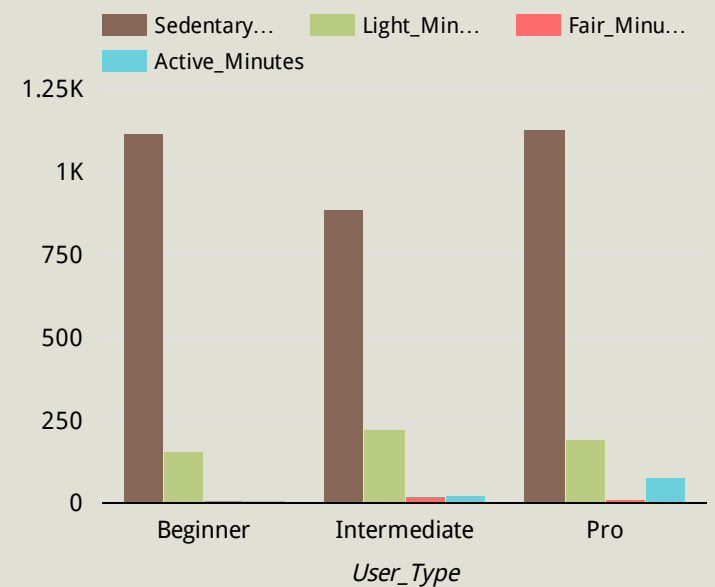
User Distance Average



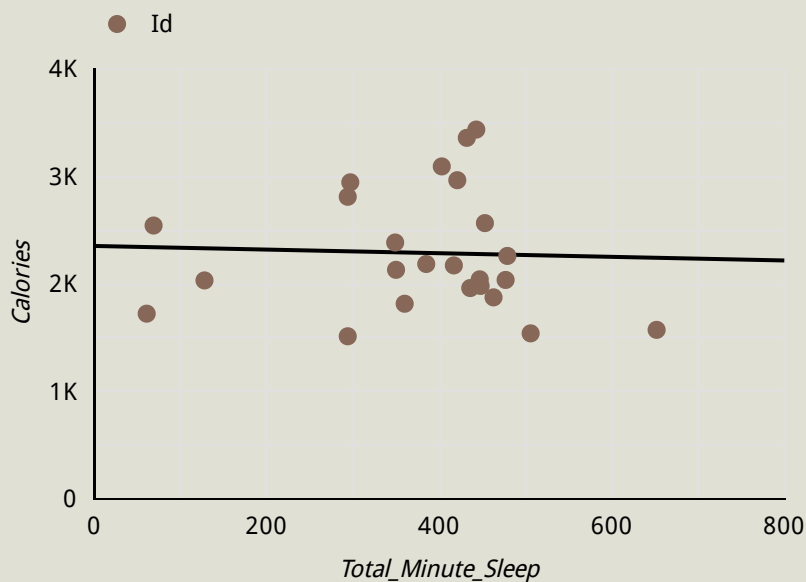
User Activity by Day



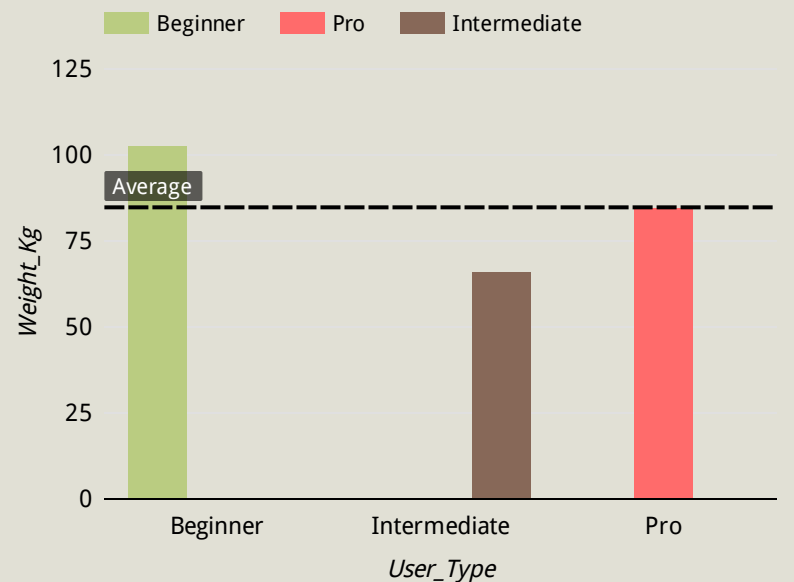
Intensity Minutes by User Type



Calories and Total Minute Sleep relation



Average Weight



From this table we only got 8 user type: Beginner 2, Intermediate 5, and Pro 1.