BellaBeat_Case_Study

Htoo Pyae Shan

2025-06-12

###Business Task

As a junior data analyst, I have been asked to analyze consumers' data on the use of smart devices for health and wellness. The goal is to identify the trends on the use of non-Bellabeat smart devices and to apply these insights on one of the Bellabeat products. The results will include the key findings and recommendations based on the analysis.

###Data source description

The data source used for the analysis will be from 'FitBit Fitness Tracker Data' from Kaggle. The data set includes the data collected under consent through 30 FitBit users containing their daily activity, heart rate, calories burn, steps taken, sleep records and weight info. The data is stored in multiple csv files, shaped in long format and a unique id for each user is used to track their activities. While the data is creditable and original, its small sample size and collected time (2016) are introducing potential bias. These factors might affect the analysis and recommendation.

###Load Libraries

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

###Loda data

```
daily_activity_1 <- read_csv("dailyActivity_merged_3.12.16-4.11.16.csv")
```

```
## Rows: 457 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): ActivityDate
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDi...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
daily_activity_2 <- read_csv("dailyActivity_merged_4.12.16-5.12.16.csv")
```

```
## Rows: 940 Columns: 15
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): ActivityDate
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDi...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
daily_activity <-bind_rows(daily_activity_1, daily_activity_2)
head(daily_activity)
```

```
## # A tibble: 6 x 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
##        <dbl> <chr>              <dbl>         <dbl>           <dbl>
## 1 1503960366 3/25/2016         11004          7.11            7.11
## 2 1503960366 3/26/2016         17609         11.6            11.6
## 3 1503960366 3/27/2016         12736          8.53            8.53
## 4 1503960366 3/28/2016         13231          8.93            8.93
## 5 1503960366 3/29/2016         12041          7.85            7.85
## 6 1503960366 3/30/2016         10970          7.16            7.16
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

```
glimpse(daily_activity)
```

```
## Rows: 1,397
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDate             <chr> "3/25/2016", "3/26/2016", "3/27/2016", "3/28/~
## $ TotalSteps               <dbl> 11004, 17609, 12736, 13231, 12041, 10970, 122~
## $ TotalDistance            <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
## $ TrackerDistance          <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveDistance       <dbl> 2.57, 6.92, 4.66, 3.19, 2.16, 2.36, 2.29, 3.3~
## $ ModeratelyActiveDistance <dbl> 0.46, 0.73, 0.16, 0.79, 1.09, 0.51, 0.49, 0.8~
## $ LightActiveDistance      <dbl> 4.07, 3.91, 3.71, 4.95, 4.61, 4.29, 5.04, 3.6~
## $ SedentaryActiveDistance  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.0~
## $ VeryActiveMinutes        <dbl> 33, 89, 56, 39, 28, 30, 33, 47, 40, 15, 43, 3~
## $ FairlyActiveMinutes      <dbl> 12, 17, 5, 20, 28, 13, 12, 21, 11, 30, 18, 18~
## $ LightlyActiveMinutes     <dbl> 205, 274, 268, 224, 243, 223, 239, 200, 244, ~
## $ SedentaryMinutes         <dbl> 804, 588, 605, 1080, 763, 1174, 820, 866, 636~
## $ Calories                 <dbl> 1819, 2154, 1944, 1932, 1886, 1820, 1889, 186~
```

Update proper date format

```
daily_activity <- daily_activity %>%
  mutate(ActivityDate = as.Date(ActivityDate, format = "%m/%d/%Y"))
head(daily_activity)
```

```
## # A tibble: 6 x 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
##        <dbl> <date>             <dbl>         <dbl>           <dbl>
```

```
## 1 1503960366 2016-03-25        11004          7.11           7.11
## 2 1503960366 2016-03-26        17609         11.6           11.6
## 3 1503960366 2016-03-27        12736          8.53           8.53
## 4 1503960366 2016-03-28        13231          8.93           8.93
## 5 1503960366 2016-03-29        12041          7.85           7.85
## 6 1503960366 2016-03-30        10970          7.16           7.16
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

###Remove duplicates

```
daily_activity <- daily_activity %>%
  distinct()
```

###Check null values

```
colSums(is.na(daily_activity))
```

```
##                       Id             ActivityDate               TotalSteps
##                        0                        0                        0
##            TotalDistance          TrackerDistance LoggedActivitiesDistance
##                        0                        0                        0
##       VeryActiveDistance ModeratelyActiveDistance      LightActiveDistance
##                        0                        0                        0
##  SedentaryActiveDistance        VeryActiveMinutes      FairlyActiveMinutes
##                        0                        0                        0
##     LightlyActiveMinutes         SedentaryMinutes                 Calories
##                        0                        0                        0
```

###Process Summary

I combined the two daily activity files, update 'ActivityDate' with proper date format and removed duplicated entries. There is no missing values found in the data set.

###Summarising activity levels

```
daily_activity %>%
  summarise(
    avg_very_active= mean(VeryActiveMinutes),
    avg_fairly_active= mean(FairlyActiveMinutes),
    avg_lightly_active= mean(LightlyActiveMinutes),
    avg_sedentary= mean(SedentaryMinutes)
  )
```

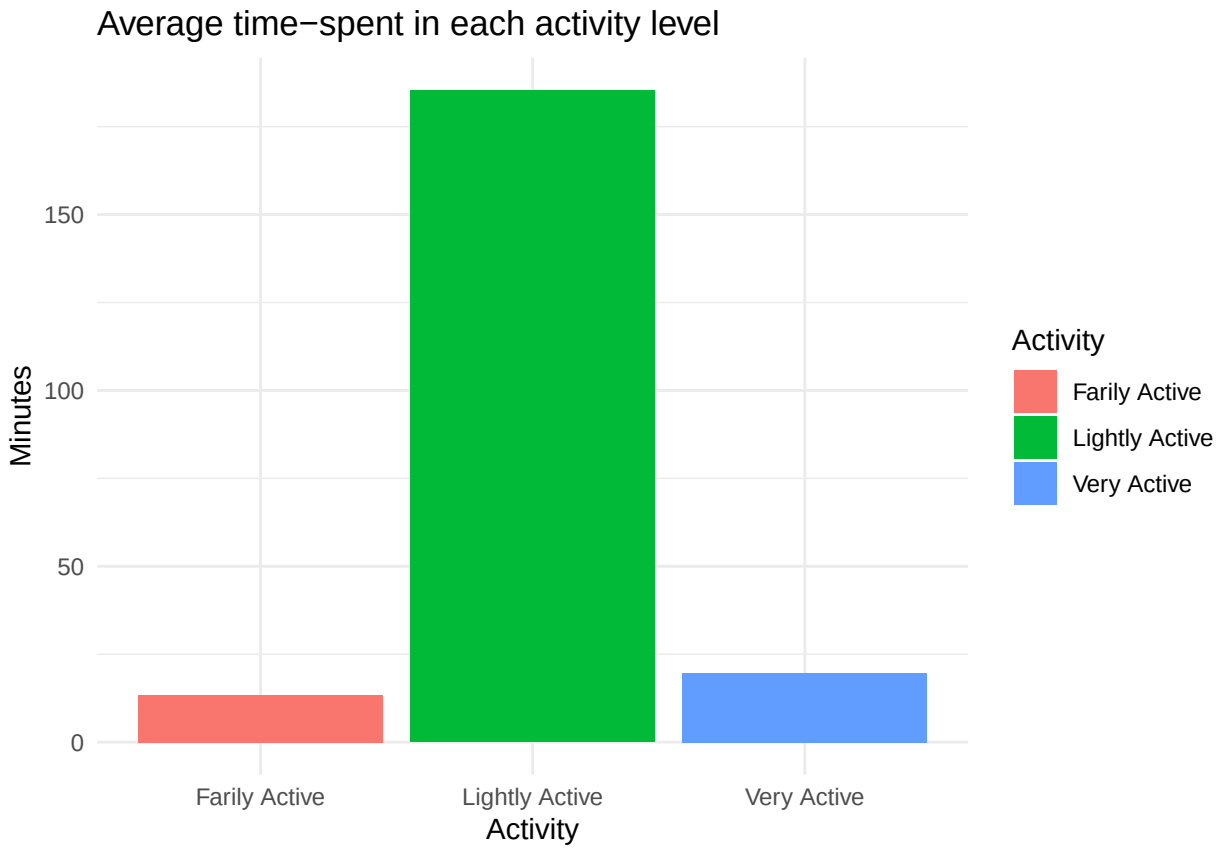
```
## # A tibble: 1 x 4
##   avg_very_active avg_fairly_active avg_lightly_active avg_sedentary
##             <dbl>             <dbl>              <dbl>         <dbl>
## 1            19.7              13.4               185.          993.
```

###Adding visuals

```
activity_summary <- daily_activity %>%
  summarise(
    "Very Active"= mean(VeryActiveMinutes),
    "Farily Active"= mean(FairlyActiveMinutes),
    "Lightly Active"= mean(LightlyActiveMinutes),
```

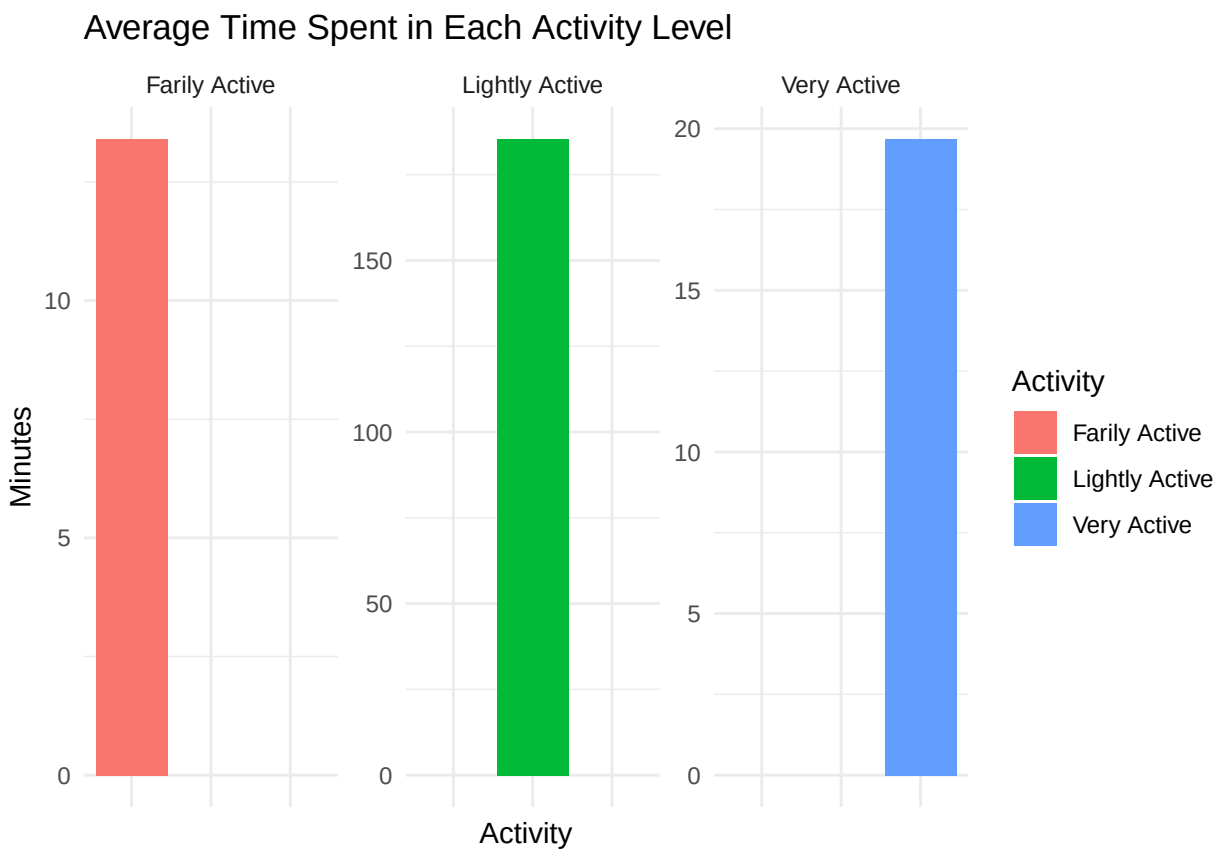
```
  )%>%
pivot_longer(cols = everything(), names_to = "Activity", values_to = "Minutes")

ggplot(activity_summary, aes(x= Activity, y= Minutes, fill= Activity))+ geom_col()+ labs(title= "Averag
```



Reshaping apprpriate Bar plot

```
ggplot(activity_summary, aes(x = Activity, y = Minutes, fill = Activity)) +
  geom_col() +
  facet_wrap(~Activity, scales = "free_y") +  # Separate y-axis for each
  labs(title = "Average Time Spent in Each Activity Level") +
  theme_minimal() +
  theme(axis.text.x = element_blank())  # Remove x-axis labels as redundant
```



###TotalSteps and BurnedCalories correlation

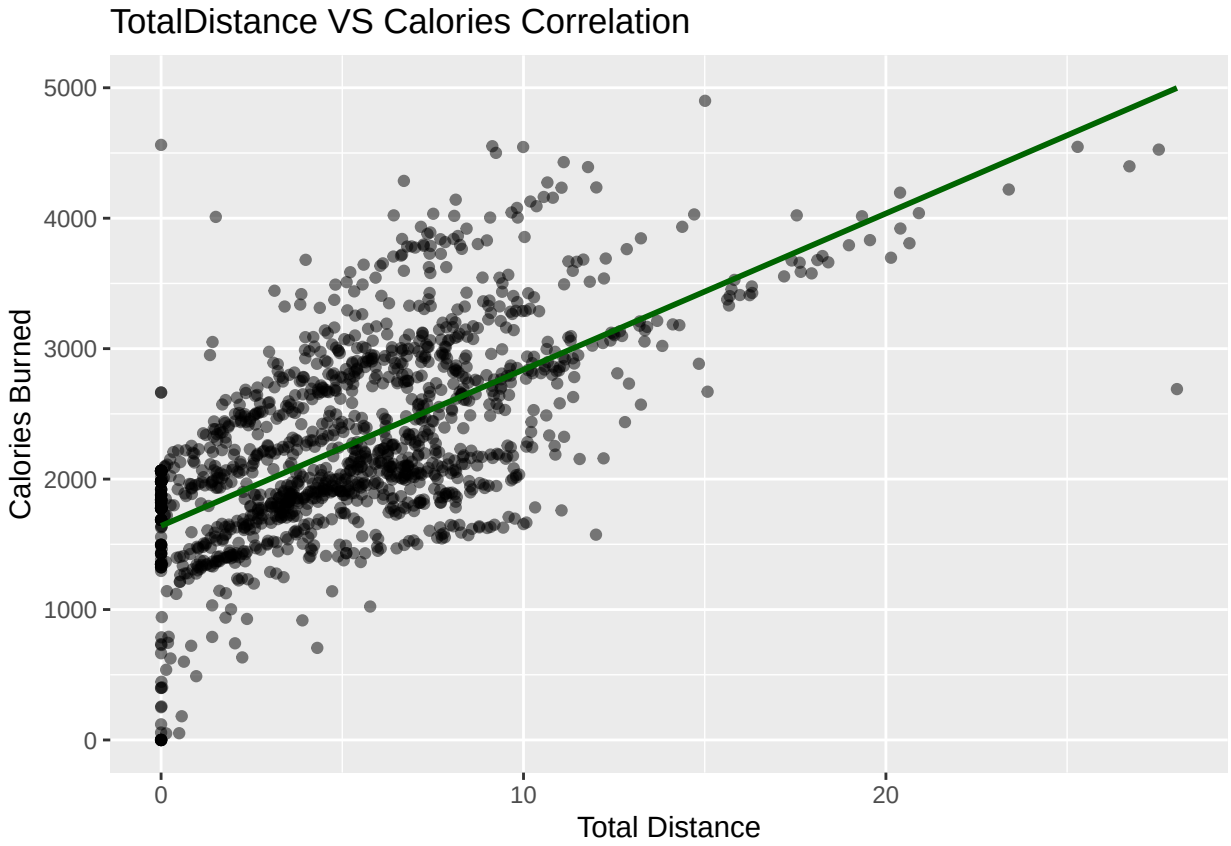
```
ggplot(daily_activity, aes(x= TotalSteps, y=Calories))+
  geom_point(alpha= 0.5)+
  geom_smooth(method = "lm", se= FALSE, color= "red")+
  labs(title = "TotalSteps VS Calories Correlation",x="Total Steps", y="Calories Burned")
```

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

###TotalDistance and BurnedCalories correlation

```
ggplot(daily_activity, aes(x= TotalDistance, y=Calories))+
  geom_point(alpha= 0.5)+
  geom_smooth(method = "lm", se= FALSE, color= "darkgreen")+
  labs(title = "TotalDistance VS Calories Correlation",x="Total Distance", y="Calories Burned")
```

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



###Key findings

Users spend the majority of their time in lightly active state. highly active minutes and steps are patently correlated with calories burned.

###Recommendation for BellaBeat

Bellabeat can use its devices (like Leaf or Time) to encourage consistent low-intensity movement by promoting daily step goals and reminders. Design features or content (like gamification or guided challenges) aimed at users with sedentary habits.