Bellabeat Analysis - Google Capstone Project

Sarath Chandrika K

Problem Statement

For this Analysis fit bit smart device data is used to analyse and provide business insights that could help to increase the sales and to unlock new growth opportunities for Bellabeat Smart Watch company.

To achieve this, the usage trend of various Bellabeat smart products used by people are collected, analysed and insights are drawn. With these insights a business strategy can be improved accordingly.

Stakeholders in this analysis include Primary Stakeholders:

- Urska Srsen - Bellabeat's Co Founder and Chief Creative Officer
- Sando Mur - Mathematician and Bellabeats cofounder

Secondary Stakeholders:

- Bellabeat Marketing Analytics Team

Data Source

data source. Data is collected from Kaggle. FitBit Fitness Tracker Data. It contains personal fitness tracker data from 30 users. It contains 18 csv files that includes various data such as daily activity, daily calories, daily steps, heart rate, hourly calories, sleep and weight data. All this information is used to analyse and solve the problem statement described above. Data is gathered in monthly, weekly and hourly format based on the id assigned to each person. These datasets were generated by respondents to a distributed survey via Amazon Mechanical Turk between 03.12.2016-05.12.2016. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring.

Data Credibility

```
To check data credibility ROCCC parameters that defines a good data set can be used.
```

1. Reliable - Data source is not reliable since it has collective data of only 30 participants which is a huge limitation for data analysis. It's kind of biased and doesn't represent the whole population.
2. Original - Data set is collected from a survey via Amazon mechanical turk leading to second or third party information, concluding that dataset is not original.
3. Comprehensive - Given dataset is not comprehensive. Most of the info to solve the problem statement is missing. No information about gender, age is mentioned in the data. This could lead to less accurate conclusions during the analysis part.
4. Current - Dataset is not current. It is from 2016 and it may not be used efficiently to come up with a business strategy now.
5. Cited - Dataset is not cited. Just the name of the survey is mentioned. It's difficult to confirm whether it's a credible source or not.

Data Storage

All the collected data is stored in spreadsheets. Being around 18 files of data, it is better to preprocess data from R instead of spreadsheets alone. Data cleaning steps are executed using different packages as mentioned below. After data pre-processing, modified spreadsheets are used for analyzing and creating visualizations.

Loading csv files

```
dailyactivity <-
  read.csv("Fitabase Data 4.12.16-5.12.16/dailyActivity_merged.csv")
dailycalories <-
  read.csv("Fitabase Data 4.12.16-5.12.16/dailyCalories_merged.csv")
dailyintensities <-
  read.csv("Fitabase Data 4.12.16-5.12.16/dailyIntensities_merged.csv")
dailysteps <-
  read.csv("Fitabase Data 4.12.16-5.12.16/dailySteps_merged.csv")
heartrate_second <-
  read.csv("Fitabase Data 4.12.16-5.12.16/heartrate_seconds_merged.csv")
hourlycalories <-
  read.csv("Fitabase Data 4.12.16-5.12.16/hourlyCalories_merged.csv")
hourlyintensities <-
  read.csv("Fitabase Data 4.12.16-5.12.16/hourlyIntensities_merged.csv")
hourlySteps <-
  read.csv("Fitabase Data 4.12.16-5.12.16/hourlySteps_merged.csv")
minutecalories_narrow <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteCaloriesNarrow_merged.csv")
minutecalories_wide <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteCaloriesWide_merged.csv")
minuteintensities_narrow <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteIntensitiesNarrow_merged.csv")
minuteintensities_wide <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteIntensitiesWide_merged.csv")
minutemets_narrow <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteMETsNarrow_merged.csv")
minutesleep <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteSleep_merged.csv")
minutessteps_narrow <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteStepsNarrow_merged.csv")
minutesteps_wide <-
  read.csv("Fitabase Data 4.12.16-5.12.16/minuteStepsWide_merged.csv")
sleepday <-
  read.csv("Fitabase Data 4.12.16-5.12.16/sleepDay_merged.csv")
weightlog <-
  read.csv("Fitabase Data 4.12.16-5.12.16/weightLogInfo_merged.csv")
```

R packages

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(tidyr)
library(stringr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

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

## v tibble  3.1.2     v purrr   0.3.4
## v readr   1.4.0     v forcats 0.5.1

## Warning: package 'forcats' was built under R version 4.1.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

```
library(hrbrthemes)
```

```
## Warning: package 'hrbrthemes' was built under R version 4.1.1

## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(viridis)
```

```
## Warning: package 'viridis' was built under R version 4.1.1
```

```
## Loading required package: viridisLite
```

```
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
library(ggcorrplot)
```

Data Summary

After observing all the data files, out of 18 files only 4 files seem to be used for Analysis. These 4 files contains data for the whole day. Rest of the files have hour, minute data which might not play an important role. Following are the four files with dataframe names

1. dailyActivity_merged.csv - dailyactivity
2. Heartrate_seconds_merged.csv - heartrate_second
3. sleepDay_merged.csv - sleepday
4. weightLogInfo_merged.csv - weightlog

Data Preparation

Splitting Date and Time into separate columns in dataframes

```
heartrate_second<-
  heartrate_second %>%
  separate(Time, c("Date", "Time"), " ")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 2483658 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
sleepday <-
  sleepday %>%
  separate(SleepDay, c("Date", "Time"), " ")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 413 rows [1, 2, 3, 4,
## 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
weightlog <-
  weightlog %>%
  separate(Date, c("Date", "Time"), " ")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 67 rows [1, 2, 3, 4,
## 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

Calculating average heartbeat in a day for each person

```
heartbeat_daily <-
  tibble(heartrate_second %>%
           group_by(Date, Id) %>%
           summarise(MeanHeartBeat=(mean(Value))))
```

```
## ‘summarise()‘ has grouped output by ’Date’. You can override using the ‘.groups‘ argument.
```

Dividing and grouping heart data time into morning, afternoon, evening, night

```
  heartrate_time <-
    read_csv("Fitabase Data 4.12.16-5.12.16/heartrate_seconds_merged.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Id = col_double(),
##   Time = col_character(),
##   Value = col_double()
## )
```

```
  heartrate_time$time <- dmy_hms(heartrate_time$Time)
```

```
## Warning: 1491097 failed to parse.
```

```
  heartrate_time <- na.omit(heartrate_time)
  breaks <- hour(hm("6:00", "12:00", "16:00", "19:00", "23:59"))
  labels <- c("Morning", "Afternoon", "Evening", "Night")
  heartrate_time$Time_of_day <- cut(x=hour(heartrate_time$time), breaks = breaks, labels = labels,
  heartrate_time <- heartrate_time %>% drop_na()
```

Grouping

```
heartbeat_grouping <-
  tibble(heartrate_time %>%
           group_by(Time_of_day) %>%
           summarise(MeanValue=(mean(Value))))
heartbeat_grouping <- heartbeat_grouping %>% drop_na()
```

Finding duplicates in each data frame

```
nrow(dailyactivity[duplicated(dailyactivity),])
```

```
## [1] 0
```

```
nrow(heartbeat_daily[duplicated(heartbeat_daily),])
```

```
## [1] 0
```

```
nrow(sleepday[duplicated(sleepday),])
```

```
## [1] 3
```

```
nrow(weightlog[duplicated(weightlog),])
```

```
## [1] 0
```

Removing duplicates from sleepday dataframe

```
sleepdata <- dplyr::distinct(sleepday)
```

Finding null values in dataframes

```
which(is.na(dailyactivity))
```

```
## integer(0)
```

```
which(is.na(heartbeat_daily))
```

```
## integer(0)
```

```
which(is.na(sleepday))
```

```
## integer(0)
```

```
which(is.na(weightlog))
```

```
##  [1] 337 338 339 340 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356
## [20] 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375
## [39] 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394
## [58] 395 396 397 398 399 400 401 402
```

Most of the values in fat column of weight log are Null, so removing that column

```
weightlog <- select(weightlog, -Fat)
```

Creating a data frame with common users from all the data frames

```
combined_df <- merge(dailyactivity, sleepday, by.x=c("Id", "ActivityDate"), by.y=c("Id", "Date"))
combined_df <- merge(combined_df, weightlog, by.x=c("Id", "ActivityDate"), by.y=c("Id", "Date"))
combined_df <- merge(combined_df, heartbeat_daily, by.x=c("Id", "ActivityDate"), by.y=c("Id", "Date"))
```

Data Analysis

Analysis is done based on the modified data frames. By the summary of data in all the data frames following are the observations, conclusions.

Dimensions of data frames

```
dim(dailyactivity)
```

```
## [1] 940  15
```

```
dim(heartbeat_daily)
```

```
## [1] 334   3
```

```
dim(sleepday)
```

```
## [1] 413   6
```

```
dim(weightlog)
```

```
## [1] 67  8
```

Unique number of persons in each data frame

```
length(unique(weightlog$Id))
```

```
## [1] 8
```

```
length(unique(heartbeat_daily$Id))
```

```
## [1] 14
```

```
length(unique(sleepday$Id))
```

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

```
length(unique(dailyactivity$Id))
```

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

```
length(unique(combined_df$Id))
```

```
## [1] 3
```

Highest of 33 participants contributed to the dataset and only 3 persons contributed data for all the features.

Data frames - Summary

```
dailyactivity %>%
  select(TotalSteps,
         TotalDistance,
         TrackerDistance,
         SedentaryMinutes,
         VeryActiveMinutes,
         FairlyActiveMinutes,
         LightlyActiveMinutes,
         Calories) %>%
  summary()
```

```
##    TotalSteps    TotalDistance    TrackerDistance  SedentaryMinutes
##  Min.   :    0   Min.   : 0.000   Min.   : 0.000   Min.   :   0.0  
##  1st Qu.: 3790   1st Qu.: 2.620   1st Qu.: 2.620   1st Qu.: 729.8  
##  Median : 7406   Median : 5.245   Median : 5.245   Median :1057.5  
##  Mean   : 7638   Mean   : 5.490   Mean   : 5.475   Mean   : 991.2  
##  3rd Qu.:10727   3rd Qu.: 7.713   3rd Qu.: 7.710   3rd Qu.:1229.5  
##  Max.   :36019   Max.   :28.030   Max.   :28.030   Max.   :1440.0  
##  VeryActiveMinutes FairlyActiveMinutes LightlyActiveMinutes    Calories   
##  Min.   :  0.00    Min.   :  0.00      Min.   :  0.0        Min.   :   0  
##  1st Qu.:  0.00    1st Qu.:  0.00      1st Qu.:127.0        1st Qu.:1828  
##  Median :  4.00    Median :  6.00      Median :199.0        Median :2134  
##  Mean   : 21.16    Mean   : 13.56      Mean   :192.8        Mean   :2304  
##  3rd Qu.: 32.00    3rd Qu.: 19.00      3rd Qu.:264.0        3rd Qu.:2793  
##  Max.   :210.00    Max.   :143.00      Max.   :518.0        Max.   :4900  
```

Observations for Daily Activity data

- Average total steps per person is 7638 per day.
- Summary of measures of tracker and total distance are same.
- Average of 2304 calories are burnt per day by a person.
- Number of people provided data to the dataset are: 33

```
sleepday %>%
  select(TotalSleepRecords,
         TotalMinutesAsleep,
         TotalTimeInBed) %>%
  summary()
```

```
##  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##  Min.   :1.000     Min.   : 58.0      Min.   : 61.0
##  1st Qu.:1.000     1st Qu.:361.0      1st Qu.:403.0
##  Median :1.000     Median :433.0      Median :463.0
##  Mean   :1.119     Mean   :419.5      Mean   :458.6
##  3rd Qu.:1.000     3rd Qu.:490.0      3rd Qu.:526.0
##  Max.   :3.000     Max.   :796.0      Max.   :961.0
```

Observations for Sleep data

- Mean, Median and Mode value of Total Sleep Records is around 1.
- Average sleeping time is 419.5 min ~ 7hr
- Average Total time in bed is 458.6 min.
- Number of people provided data to the dataset are: 24

```
summary(heartbeat_daily$MeanHeartBeat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   59.38   70.47   77.49   78.61   84.93  109.79
```

Observations for Mean Heart beat data

- Average value of heart rate is around 77
- Median value of heart rate is around 73
- Number of people provided data to the dataset are: 14

```
weightlog %>%
  select(WeightKg,
         BMI) %>%
  summary()
```

```
##     WeightKg           BMI
##  Min.   : 52.60   Min.   :21.45
##  1st Qu.: 61.40   1st Qu.:23.96
##  Median : 62.50   Median :24.39
##  Mean   : 72.04   Mean   :25.19
##  3rd Qu.: 85.05   3rd Qu.:25.56
##  Max.   :133.50   Max.   :47.54
```

Observations for Weight data

- Mean weight is 72kg.
- Mean BMI is 25.19
- Mean Fat is 23.50
- Number of people provided data to the dataset are: 8

```
heartrate_time %>%
  select(Value,
         Time_of_day) %>%
  summary()
```

```
##      Value           Time_of_day
##  Min.   : 38.0   Morning  :329894
##  1st Qu.: 66.0   Afternoon:208150
##  Median : 77.0   Evening  :150108
##  Mean   : 79.8   Night    :139066
##  3rd Qu.: 90.0
##  Max.   :199.0
```

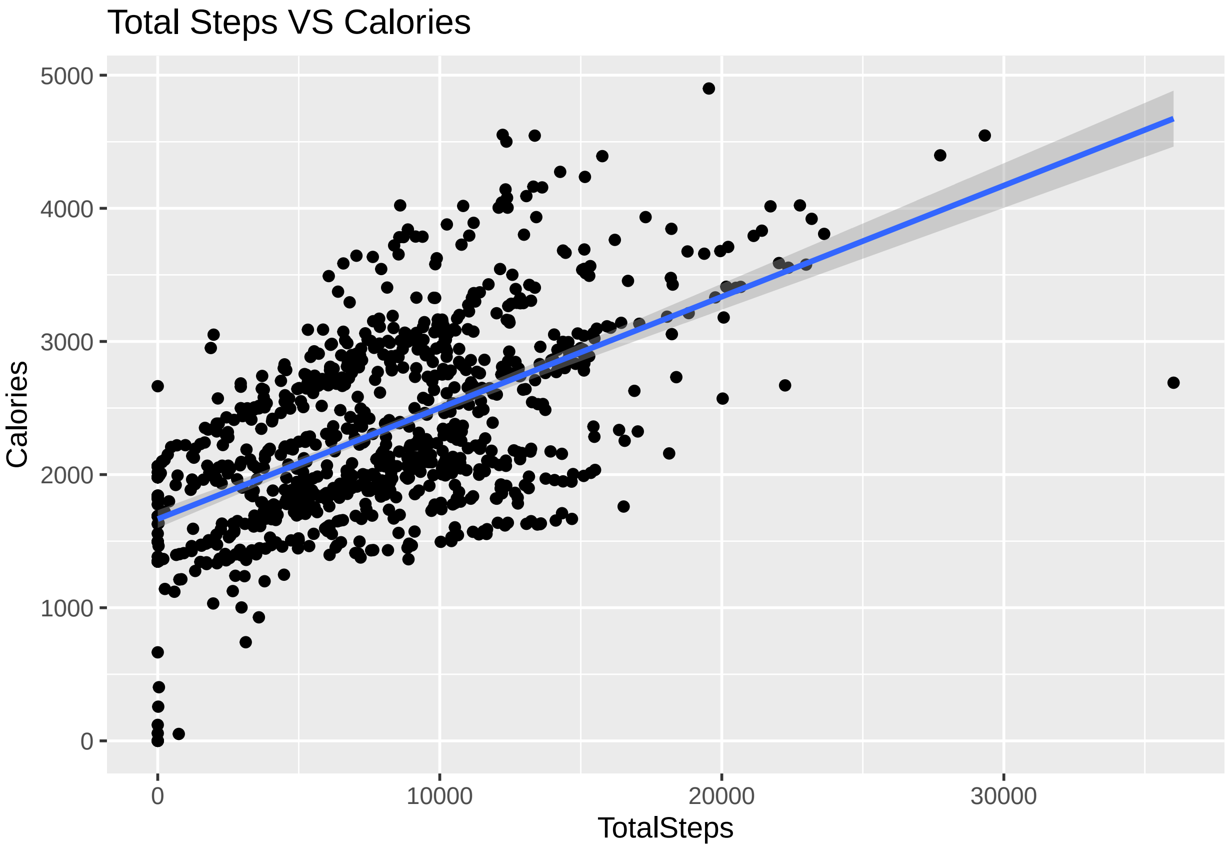
Observations for Heart rate data based on time of the day

- With respect to count most of the records are during morning.

Data Visualization

```
fig <- ggplot(data=dailyactivity, aes(x=TotalSteps, y=Calories)) +
  geom_point() +
  geom_smooth(method=lm) +
  labs(title="Total Steps VS Calories")
plot(fig)
```

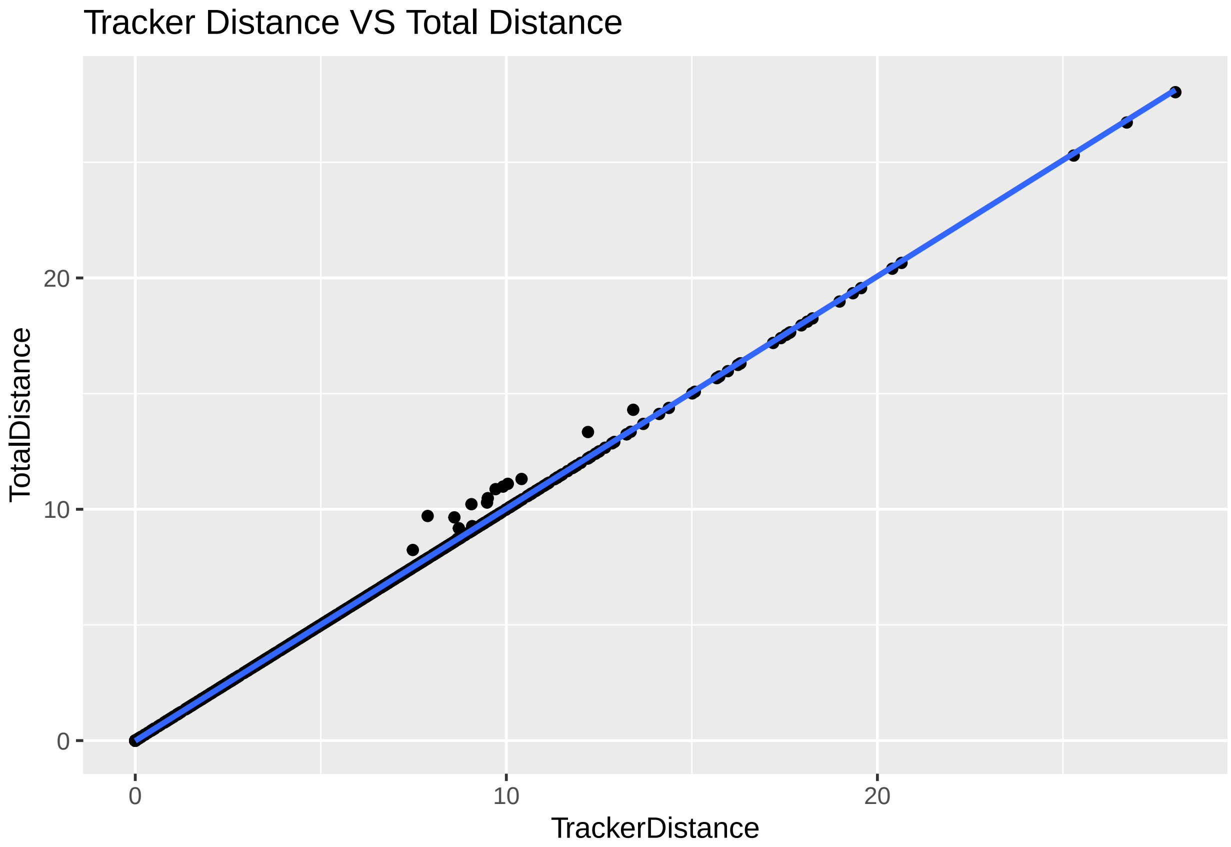
```
## 'geom_smooth()' using formula 'y ~ x'
```



From the above figure it can be seen that TotalSteps and Calories are positively correlated. As total steps increase number of calories also increase. In few cases even if total steps are not high, calories burnt are high.

```
ggplot(data=dailyactivity, aes(x=TrackerDistance, y=TotalDistance)) +
  geom_point() +
  geom_smooth(method = lm) +
  labs(title = "Tracker Distance VS Total Distance")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



Total distance and Tracker Distance are almost same. This depicts that Bellabeat smart watch is accurate in calculating the distance.

```
ggplot(data=dailyactivity, aes(x=VeryActiveMinutes , y=Calories)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Very Active Minutes VS Calories")
```

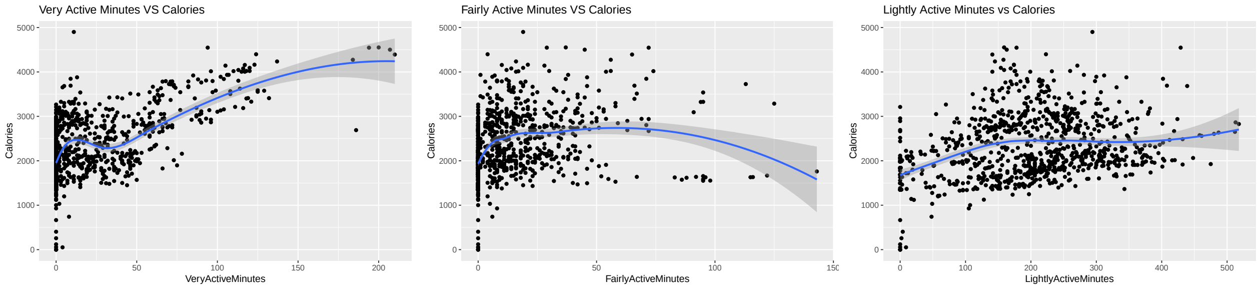
```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```

```
ggplot(data=dailyactivity, aes(x=FairlyActiveMinutes, y=Calories)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Fairly Active Minutes VS Calories")
```

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

```
ggplot(data=dailyactivity, aes(x=LightlyActiveMinutes, y=Calories)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Lightly Active Minutes vs Calories")
```

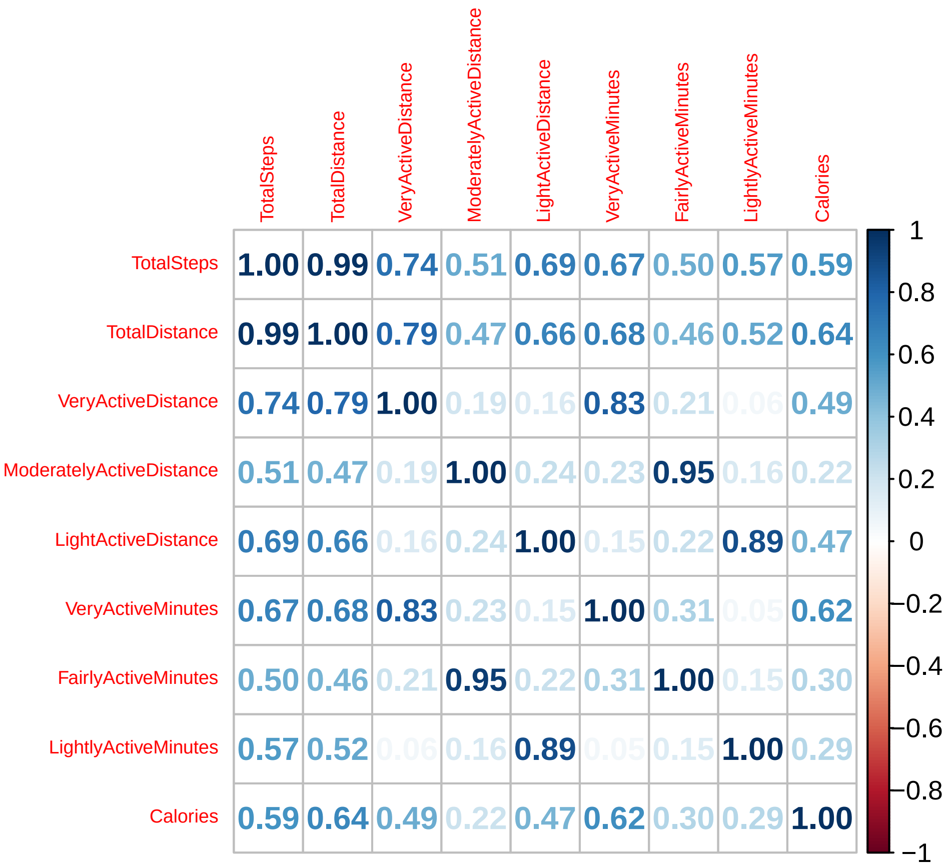
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



- From the above three graphs among Lightly Active Minutes, Fairly Active Minutes, Very Active Minutes VS Calories it is clear that fairly active minutes is negatively correlated to calories.
- Distribution of calories is more around Lightly Active Minutes.
- Most of the distribution points of very active minutes is around 0.

Correlation Matrix

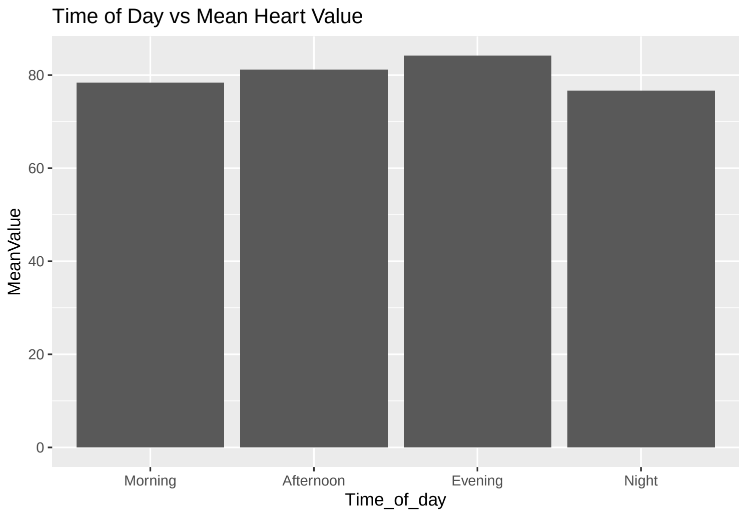
```
selected_columns <- select(dailyactivity, TotalSteps, TotalDistance, VeryActiveDistance, ModeratelyActi
corr = cor(selected_columns)
corrplot(corr, method = 'number', tl.cex = 0.6)
```



	TotalSteps	TotalDistance	VeryActiveDistance	ModeratelyActiveDistance	LightActiveDistance	VeryActiveMinutes	FairlyActiveMinutes	LightlyActiveMinutes	Calories
TotalSteps	1.00	0.99	0.74	0.51	0.69	0.67	0.50	0.57	0.59
TotalDistance	0.99	1.00	0.79	0.47	0.66	0.68	0.46	0.52	0.64
VeryActiveDistance	0.74	0.79	1.00	0.19	0.16	0.83	0.21	0.06	0.49
ModeratelyActiveDistance	0.51	0.47	0.19	1.00	0.24	0.23	0.95	0.16	0.22
LightActiveDistance	0.69	0.66	0.16	0.24	1.00	0.15	0.22	0.89	0.47
VeryActiveMinutes	0.67	0.68	0.83	0.23	0.15	1.00	0.31	0.05	0.62
FairlyActiveMinutes	0.50	0.46	0.21	0.95	0.22	0.31	1.00	0.15	0.30
LightlyActiveMinutes	0.57	0.52	0.06	0.16	0.89	0.05	0.15	1.00	0.29
Calories	0.59	0.64	0.49	0.22	0.47	0.62	0.30	0.29	1.00

Total Steps, Total Distance, Very Active Minutes have high correlation values with Calories.

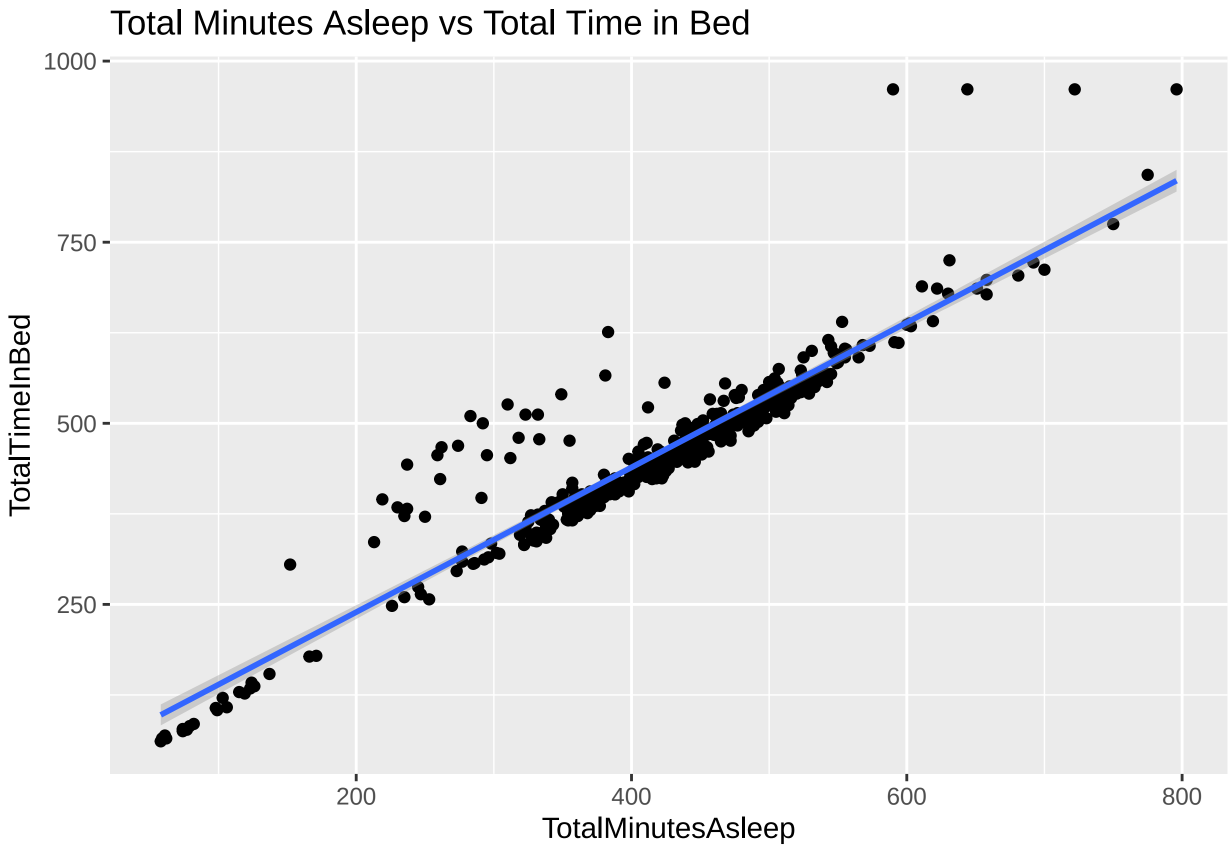
```r
ggplot(data = heartbeat_grouping, aes(x=Time_of_day, y=MeanValue)) +
  geom_bar(stat = "identity") +
  labs(title="Time of Day vs Mean Heart Value")
```



Even though most of the heart beat values recorded were in the morning, average heartbeat is low during night around 7pm - 12pm.

```
ggplot(data=sleepday, aes(x=TotalMinutesAsleep, y=TotalTimeInBed)) +
  geom_point() +
  geom_smooth(method=lm) +
  labs(title="Total Minutes Asleep vs Total Time in Bed")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



```
ggplot(data=combined_df, aes(x=VeryActiveMinutes, y=TotalMinutesAsleep)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Very Active Minutes vs Total Minutes Asleep")
```

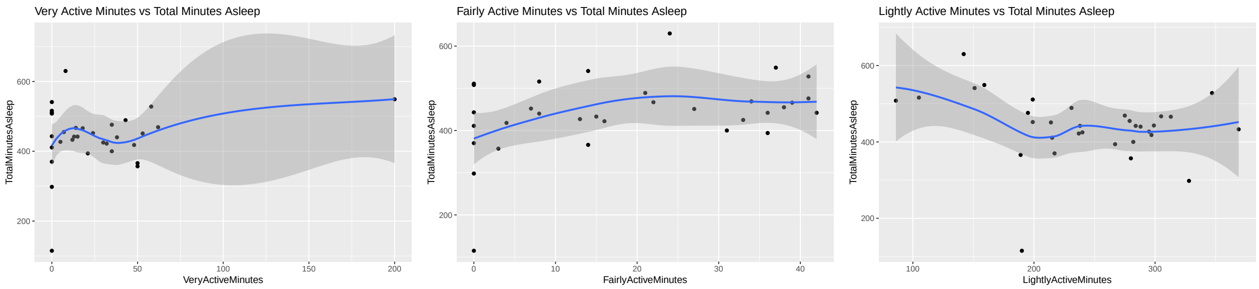
```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```

```
ggplot(data=combined_df, aes(x=FairlyActiveMinutes, y=TotalMinutesAsleep)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Fairly Active Minutes vs Total Minutes Asleep")
```

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

```
ggplot(data=combined_df, aes(x=LightlyActiveMinutes, y=TotalMinutesAsleep)) +
  geom_point() +
  geom_smooth(orientation = "x") +
  labs(title = "Lightly Active Minutes vs Total Minutes Asleep")
```

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



Conclusions from Data Analysis and Data Visualization

- Mean and median of heart rate value are very near to actual heart rate value.
- No big difference is identified between total time in bed and total sleeping time which is a good point in terms of people's health and average sleeping time is around 7hr. It can be concluded that most of the people turn out to have sufficient sleep.
- Comparing the mean of BMI to the ideal BMI range for normal weight status which is 18.5 - 24.9, it can be assumed that on an average most people are in normal range.
- Bellabeat step prediction function has turned out to be most efficient since the tracker measures and total measures are closely same.
- Number of people who contributed to the dataset is low and not all 30 members data have been collected in all the parameters/features.
- Count of people who contributed data to complete features is 3 which is pretty low and is difficult to draw analysis from this.

Conclusions for Bellabeat Marketing Analytics Team

- Based on daily activity correlation company can improve on sending push notifications for reminding to be active in frequent intervals, as movement such as total steps, total distance tend to burn more calories.
- A feature can be developed to set minimum movement target and monitoring it on timely basis.
- Maximum heartbeat is around 200 which is pretty high, so company can develop few alerts to the users based on abnormal heart rate change excluding the conditions of very active minutes.
- To continue with the same pace of total sleep time of 7hr, company can develop remainder feature for bed time based on the sleeping time of individual.

In terms of additional data, age, gender of the individual is not mentioned which play an important role. It would be better to analyse if data is collected for all the features from everyone since there are only 3 people contributing to complete dataset.