Google Capstone Project 2 – Bellabeat Case Study

Fabian Pang

2/6/2022

Introduction

This is my attempt at the Google Data Analytics Capstone - Case Study 2 (Bellabeat Case Study). The full details to the case study can be found in the course page Google Data Analytics Capstone: Complete a Case Study ().

This case study will adopt the framework as suggested by the course involving these 6 steps: Ask, Prepare, Process, Analyse, Share and Act

How can a Wellness Technology Company Play It Smart?

Ask

Background information

Bellabeat is a high-tech company that manufactures health-focused smart products targeting women. Collecting data on activity, sleep, stress, and reproductive health has allowed Bellabeat to empower women with knowledge about their own health and habits. Since it was founded in 2013, Bellabeat has grown rapidly and quickly positioned itself as a tech-driven wellness company for women.

Sršen, one of Bellabeat's founders, knows that an analysis of Bellabeat's available consumer data would reveal more opportunities for growth.

Business Task

Analyse FitBit Fitness Tracker Data to gain insights on how consumers are using the FitBit app and discover trends and insights for Bellabeat's marketing strategy

Stakeholders

Urška Sršen: Bellabeat's cofounder and Chief Creative Officer

Sando Mur: Mathematician and Bellabeat's cofounder; key member of the Bellabeat executive team

Bellabeat marketing analytics team: A team of data analysts responsible for collecting, analyzing, and reporting data that helps guide Bellabeat's marketing strategy.

Business Objectives

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?

Key deliverables

1. A clear summary of the business task
2. A description of all data sources used
3. Documentation of any cleaning or manipulation of data
4. A summary of your analysis
5. Supporting visualizations and key findings
6. Your top high-level content recommendations based on your analysis

Prepare

Information on Data

30 eligible Fitbit users consented to the submission of personal tracker data.

The data includes information about the respondents' minute-level output for physical activity, heart rate, sleep monitoring, daily activity and steps.

This dataset was generated in a distributed survey via Amazon Mechanical Turk between 12 March 2016 to 12 May 2016 and is publicly available on Kaggle (https://www.kaggle.com/arashnic/fitbit).

The dataset is organised into 18 separate csv files according to the nature of data collected.

Limitations of Data

Data was collected in 2016. User habits surrounding daily activities, physical activities, sleep habits may have changed since then, rendering the data irrelevant.

Its small sample size of 30 respondents might not be representative of the entire population of women

Data was collected in a 2 months period. User habits may vary throughout the year due to certain reasons (e.g. weather) and hence might not be entirely representative.

Key characteristics of participants were not made known (e.g. gender, age, location)

Some records are also missing

Is the Data ROCCC (Reliable, Original, Comprehensive, Current, Cited)?

Reliable Due to the limitations of the data, its reliability may not be well established.

Original No, data was collected from a third-party source (Amazon Mechanical Turk).

Comprehensive Data collected matches the parameters that Bellabeat is interested in.

Current No, data was collected 6 years ago.

Cited Data is cited

Overall, while the data may be cited and comprehensive, there are questions surrounding its credibility, potential of biasness and relevance.

Process

For this project, R will be used to prepare and process the data. For the purpose of this project, I will focus on 2 csv files, namely dailyactivities_merged and sleepDay_merged.

Loading the relevant libraries

```
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   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

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

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(lubridate)
```

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

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

```
library(dplyr)

library(skimr)

library(DescTools)

library(geosphere)

library(ggplot2)
```

Importing the relevant datasets

```
daily_activity <- read_csv("dailyActivity_merged.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_sleep <- read_csv("sleepDay_merged.csv")
```

```
## Rows: 413 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): SleepDay
## dbl (4): Id, TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed
##
## 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.
```

Inspecting the datasets

```
str(daily_activity)
```

```
## spec_tbl_df [940 x 15] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Id                       : num [1:940] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityDate             : chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016"
...
##  $ TotalSteps               : num [1:940] 13162 10735 10460 9762 12669 ...
##  $ TotalDistance            : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...
##  $ TrackerDistance          : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...
##  $ LoggedActivitiesDistance: num [1:940] 0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveDistance       : num [1:940] 1.88 1.57 2.44 2.14 2.71 ...
##  $ ModeratelyActiveDistance: num [1:940] 0.55 0.69 0.4 1.26 0.41 ...
##  $ LightActiveDistance      : num [1:940] 6.06 4.71 3.91 2.83 5.04 ...
##  $ SedentaryActiveDistance : num [1:940] 0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveMinutes        : num [1:940] 25 21 30 29 36 38 42 50 28 19 ...
##  $ FairlyActiveMinutes      : num [1:940] 13 19 11 34 10 20 16 31 12 8 ...
##  $ LightlyActiveMinutes     : num [1:940] 328 217 181 209 221 164 233 264 205 211 ...
##  $ SedentaryMinutes         : num [1:940] 728 776 1218 726 773 ...
##  $ Calories                 : num [1:940] 1985 1797 1776 1745 1863 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   ActivityDate = col_character(),
##   ..   TotalSteps = col_double(),
##   ..   TotalDistance = col_double(),
##   ..   TrackerDistance = col_double(),
##   ..   LoggedActivitiesDistance = col_double(),
##   ..   VeryActiveDistance = col_double(),
##   ..   ModeratelyActiveDistance = col_double(),
##   ..   LightActiveDistance = col_double(),
##   ..   SedentaryActiveDistance = col_double(),
##   ..   VeryActiveMinutes = col_double(),
##   ..   FairlyActiveMinutes = col_double(),
##   ..   LightlyActiveMinutes = col_double(),
##   ..   SedentaryMinutes = col_double(),
##   ..   Calories = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
str(daily_sleep)
```

```
## spec_tbl_df [413 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Id                : num [1:413] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ SleepDay          : chr [1:413] "4/12/2016 12:00:00 AM" "4/13/2016 12:00:00 AM" "4/15/2
016 12:00:00 AM" "4/16/2016 12:00:00 AM" ...
##  $ TotalSleepRecords : num [1:413] 1 2 1 2 1 1 1 1 1 ...
##  $ TotalMinutesAsleep: num [1:413] 327 384 412 340 700 304 360 325 361 430 ...
##  $ TotalTimeInBed    : num [1:413] 346 407 442 367 712 320 377 364 384 449 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   SleepDay = col_character(),
##   ..   TotalSleepRecords = col_double(),
##   ..   TotalMinutesAsleep = col_double(),
##   ..   TotalTimeInBed = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Data Cleaning and Manipulation

I would go through a series of steps to clean and manipulate the datasets to prepare them for analysis.

Removing Duplicates

Firstly, I would check for and remove duplicates in the datasets (if any).

```
## Checking for duplicates based on ID and ActivityDate in daily_activity
duplicated(daily_activity[,1:2])
```

```
##   [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [97] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [193] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [205] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [229] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [241] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [253] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [265] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [277] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [289] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [301] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [313] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [325] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [337] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [349] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [361] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [373] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [385] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [397] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [409] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [421] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [433] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [445] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [457] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [469] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [481] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [493] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [505] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [517] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [529] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [541] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [553] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [565] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [577] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [589] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [601] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [613] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [625] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [637] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [649] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
## [661] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [673] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [685] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [697] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [709] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [721] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [733] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [745] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [757] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [769] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [781] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [793] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [805] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [817] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [829] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [841] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [853] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [865] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [877] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [889] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [901] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [913] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [925] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [937] FALSE FALSE FALSE FALSE
```

```
## No duplicates in daily_activity

## Checking for duplicates based on ID and SleepDay in daily_sleep

## Removing duplicates from daily_sleep
dup <- duplicated(daily_sleep[,1:2]) # 3 duplicates


daily_sleep <- daily_sleep[!dup,]
```

Check for any Missing Values

Next, check if there are any missing values

```
sum(is.na(daily_activity))
```

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

```
## no missing values in daily_activity

sum(is.na(daily_sleep))
```

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

```
## no missing values in daily_sleep
```

Check if data makes sense

Next, check if there are “illogical” data entries. (For instance, ones where total distance = 0)

```
## Check if there are entries with total distance = 0
sum(daily_activity$TotalDistance =="0")
```

```
## [1] 78
```

```
## Remove entries with 0 for total distance

daily_activity <- daily_activity %>%
  filter(daily_activity$TotalDistance != "0")

## Check if there are entries with total steps = 0
sum(daily_activity$TotalSteps =="0")
```

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

```
## Check if there are entries with total calories = 0
sum(daily_activity$Calories =="0")
```

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

```
## Check if there are entries with total minutes asleep = 0
sum(daily_sleep$TotalMinutesAsleep =="0")
```

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

Check for number of unique IDs

```
uniqueid <- count(unique(daily_activity[,1]))
print(paste0("no of unique IDs:", uniqueid))
```

```
## [1] "no of unique IDs:33"
```

```
## 33 respondents rather than 30 as stated

uniqueids <- count(unique(daily_sleep[,1]))
print(paste0("no of unique IDs:", uniqueids))
```

```
## [1] "no of unique IDs:24"
```

```
## only 24 respondents, may have missing records
```

There are 33 respondents in the daily_activity dataset, rather than 30 as stated. On the other hand, daily_sleep only records 24 respondents which indicates that there may be missing records.

Converting data type for both ActivityDate and SleepDay

```
class(daily_activity$ActivityDate)
```

```
## [1] "character"
```

```
class(daily_sleep$SleepDay)
```

```
## [1] "character"
```

Both ActivityDate and SleepDay are of the wrong data type and have to be converted to Datetime.

```
daily_activity$ActivityDate <- as.POSIXct(daily_activity$ActivityDate, format = "%m/%d/%Y")

class(daily_activity$ActivityDate)
```

```
## [1] "POSIXct" "POSIXt"
```

```
daily_sleep$SleepDay <- as.POSIXct(daily_sleep$SleepDay, format="%m/%d/%Y %H:%M:%S")

class(daily_sleep$SleepDay)
```

```
## [1] "POSIXct" "POSIXt"
```

Now, new columns would be created for subsequent data analysis.

Creating new column DayofWeek for both datasets

Creating new columns for both datasets to extract the recorded day of week.

```
daily_activity$dayofweek <- wday(daily_activity$ActivityDate, label = TRUE)

daily_sleep$dayofweek <- wday(daily_sleep$SleepDay, label = TRUE)
```

Creating new total minutes and total hours column

```
## Creating total minutes column
daily_activity$TotalMinutes
```

```
## Warning: Unknown or uninitialised column: `TotalMinutes`.
```

```
## NULL
```

```
daily_activity[,"TotalMinutes"] <- rowSums(daily_activity[,11:14])

## Creating total hours column

daily_activity$TotalHours
```

```
## Warning: Unknown or uninitialised column: `TotalHours`.
```

```
## NULL
```

```
daily_activity[,"TotalHours"] <- (daily_activity[,"TotalMinutes"])/60
```

Creating total hours asleep column for daily_sleep

```
daily_sleep$TotalHours
```

```
## Warning: Unknown or uninitialised column: `TotalHours`.
```

```
## NULL
```

```
daily_sleep[,"TotalHours"] <- (daily_sleep[,"TotalMinutesAsleep"])/60
```

Creating total time taken to fall asleep column

```
daily_sleep$timetakentosleep
```

```
## Warning: Unknown or uninitialised column: `timetakentosleep`.
```

```
## NULL
```

```
daily_sleep[,"timetakentosleep"] <- (daily_sleep[,"TotalTimeInBed"]) - (daily_sleep[,"TotalMi
nutesAsleep"])
```

Important assumption: time spent awake on bed by respondents is time spent trying to fall asleep and not for other reasons such as lazing on bed after waking up.

Data cleaning and manipulation is completed. Now, the data is ready to be analysed.

Analyse and Share

For this project, the analyse and share step would be combined together.

Descriptive Statistics

Firstly, let's take a look at some descriptive statistics

```
summary(daily_activity)
```

```
##        Id              ActivityDate                    TotalSteps
##  Min.   :1.504e+09   Min.   :2016-04-12 00:00:00   Min.   :    8
##  1st Qu.:2.320e+09   1st Qu.:2016-04-18 00:00:00   1st Qu.: 4927
##  Median :4.445e+09   Median :2016-04-26 00:00:00   Median : 8054
##  Mean   :4.861e+09   Mean   :2016-04-26 02:15:18   Mean   : 8329
##  3rd Qu.:6.962e+09   3rd Qu.:2016-05-03 00:00:00   3rd Qu.:11096
##  Max.   :8.878e+09   Max.   :2016-05-12 00:00:00   Max.   :36019
##
##  TotalDistance    TrackerDistance  LoggedActivitiesDistance VeryActiveDistance
##  Min.   : 0.010   Min.   : 0.010   Min.   :0.000            Min.   : 0.000
##  1st Qu.: 3.373   1st Qu.: 3.373   1st Qu.:0.000            1st Qu.: 0.000
##  Median : 5.590   Median : 5.590   Median :0.000            Median : 0.410
##  Mean   : 5.986   Mean   : 5.971   Mean   :0.118            Mean   : 1.639
##  3rd Qu.: 7.905   3rd Qu.: 7.880   3rd Qu.:0.000            3rd Qu.: 2.277
##  Max.   :28.030   Max.   :28.030   Max.   :4.942            Max.   :21.920
##
##  ModeratelyActiveDistance LightActiveDistance SedentaryActiveDistance
##  Min.   :0.0000           Min.   : 0.000      Min.   :0.000000
##  1st Qu.:0.0000           1st Qu.: 2.350      1st Qu.:0.000000
##  Median :0.3100           Median : 3.580      Median :0.000000
##  Mean   :0.6189           Mean   : 3.643      Mean   :0.001752
##  3rd Qu.:0.8675           3rd Qu.: 4.897      3rd Qu.:0.000000
##  Max.   :6.4800           Max.   :10.710      Max.   :0.110000
##
##  VeryActiveMinutes FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes
##  Min.   :  0.00    Min.   :  0.00      Min.   :  0.0        Min.   :   0.0
##  1st Qu.:  0.00    1st Qu.:  0.00      1st Qu.:147.0        1st Qu.: 721.2
##  Median :  7.00    Median :  8.00      Median :208.5        Median :1020.5
##  Mean   : 23.04    Mean   : 14.79      Mean   :210.3        Mean   : 955.2
##  3rd Qu.: 35.00    3rd Qu.: 21.00      3rd Qu.:272.0        3rd Qu.:1189.0
##  Max.   :210.00    Max.   :143.00      Max.   :518.0        Max.   :1440.0
##
##     Calories    dayofweek  TotalMinutes      TotalHours
##  Min.   :  52   Sun:110   Min.   :   2.0   Min.   : 0.03333
##  1st Qu.:1857   Mon:110   1st Qu.: 983.2   1st Qu.:16.38750
##  Median :2220   Tue:138   Median :1349.5   Median :22.49167
##  Mean   :2362   Wed:138   Mean   :1203.3   Mean   :20.05476
##  3rd Qu.:2832   Thu:133   3rd Qu.:1440.0   3rd Qu.:24.00000
##  Max.   :4900   Fri:120   Max.   :1440.0   Max.   :24.00000
##                 Sat:113
```

Descriptive Statistics of total distance

```
summary(daily_activity$TotalDistance)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.010   3.373   5.590   5.986   7.905  28.030
```

To note: metadata (https://www.fitabase.com/media/1930/fitabasedatadictionary102320.pdf) states that distance is measured in km.

Descriptive Statistics of total calories

```
summary(daily_activity$Calories)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      52    1857    2220    2362    2832    4900
```

Information on calories burned may not be as useful without the other key demographic information such as age, gender, weight etc.

Descriptive Statistics of total hours of sleep

```
summary(daily_sleep$TotalHours)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.9667  6.0167  7.2083  6.9862  8.1667 13.2667
```

Average amount of sleep obtained by respondents is inadequate. It is recommended that one should have at least 7-9 hours of sleep daily (Source: Centers for Disease Control and Prevention (https://www.cdc.gov/sleep/about_sleep/how_much_sleep.html)).

Relationship between Total Distance and Calories

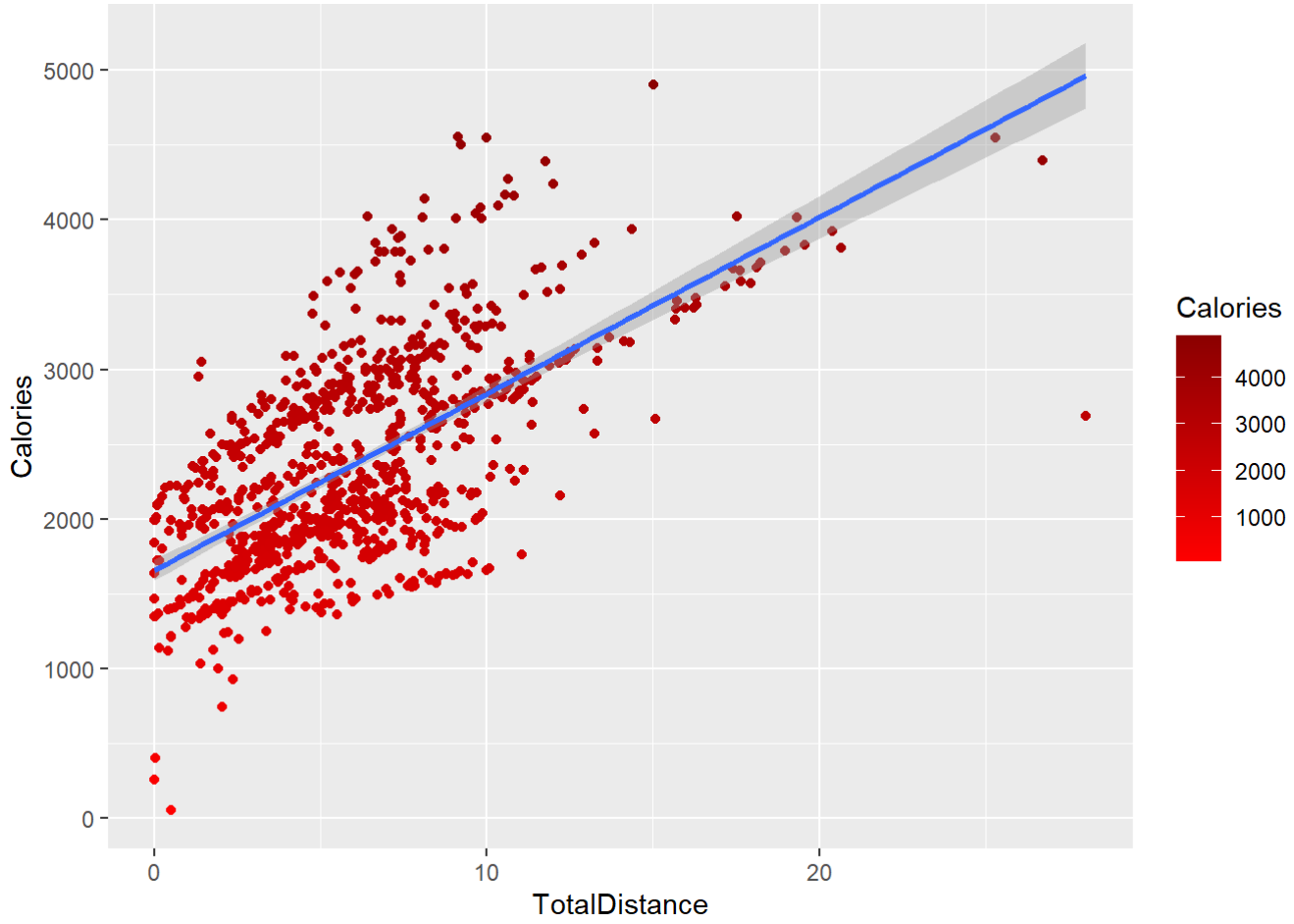
```
cor(daily_activity$TotalDistance, daily_activity$Calories, method = "pearson")
```

```
## [1] 0.624238
```

There is a positive correlation between total distance traveled and calories burned. (As total distance traveled increases, the amount of calories burned increases.)

```
ggplot(daily_activity, aes(x=TotalDistance, y=Calories, color = Calories))+
  geom_point()+
  scale_colour_gradient(low = "red", high = "darkred")+
  geom_smooth(method=lm)
```

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



Relationship between total distance traveled and calories burned is stronger from 0-10km, beyond that the strength of the relationship decreases (which suggests that amount of calories burned after 10km may not increase as proportionately)

Relationship between total steps and calories

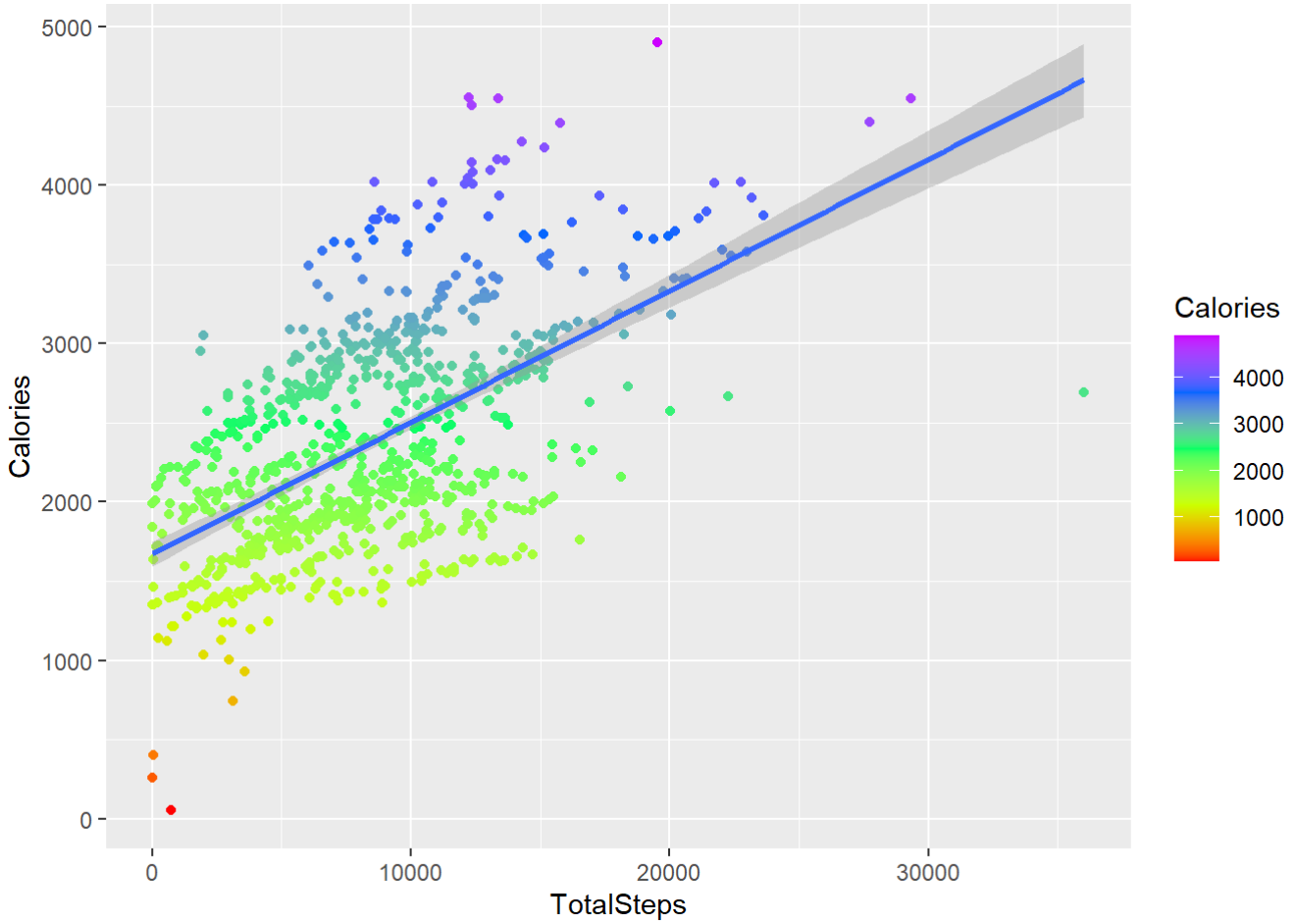
```
cor(daily_activity$TotalSteps, daily_activity$Calories, method = "pearson")
```

```
## [1] 0.5607237
```

There is a positive correlation between total steps taken and calories burned. (As total steps taken increases, the amount of calories burned increases.)

```
ggplot(daily_activity, aes(x=TotalSteps, y=Calories, color = Calories))+
  geom_point()+
  scale_colour_gradientn(colours=rainbow(5))+
  geom_smooth(method=lm)
```

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



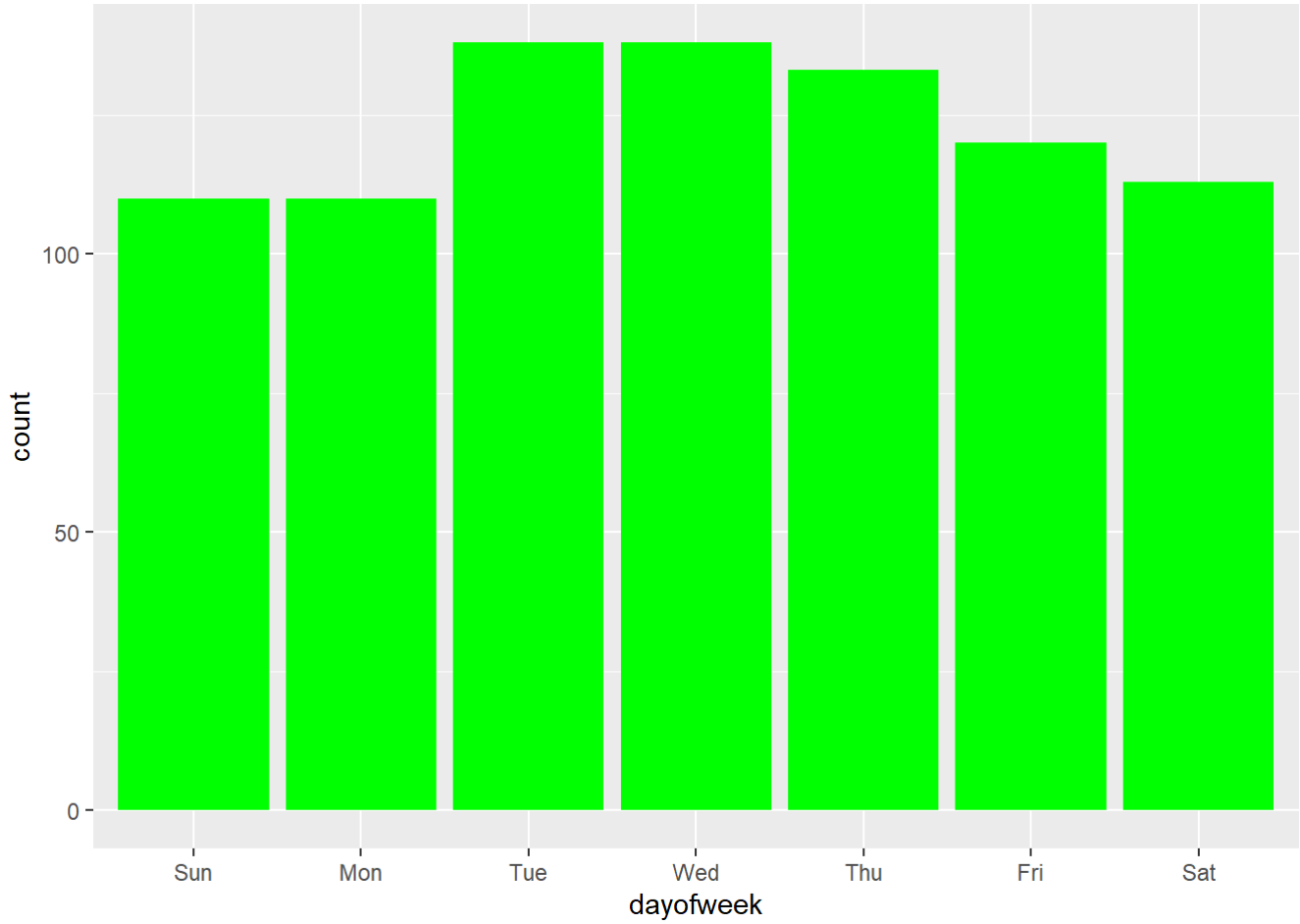
Relationship between total steps taken and calories burned is stronger from 0 - approximately < 15000 steps, beyond that the strength of the relationship decreases (which suggests that amount of calories burned after 15000 steps may not increase as proportionately). This is similar to the previous relationship between total distance traveled and calories burned.

There are several outliers present in data as well (close to 0 steps taken and 0 calories burned; >35000 steps taken but less than 3000 calories burned).

In the next part of the analysis, the variables will be analysed across the various days of the week.

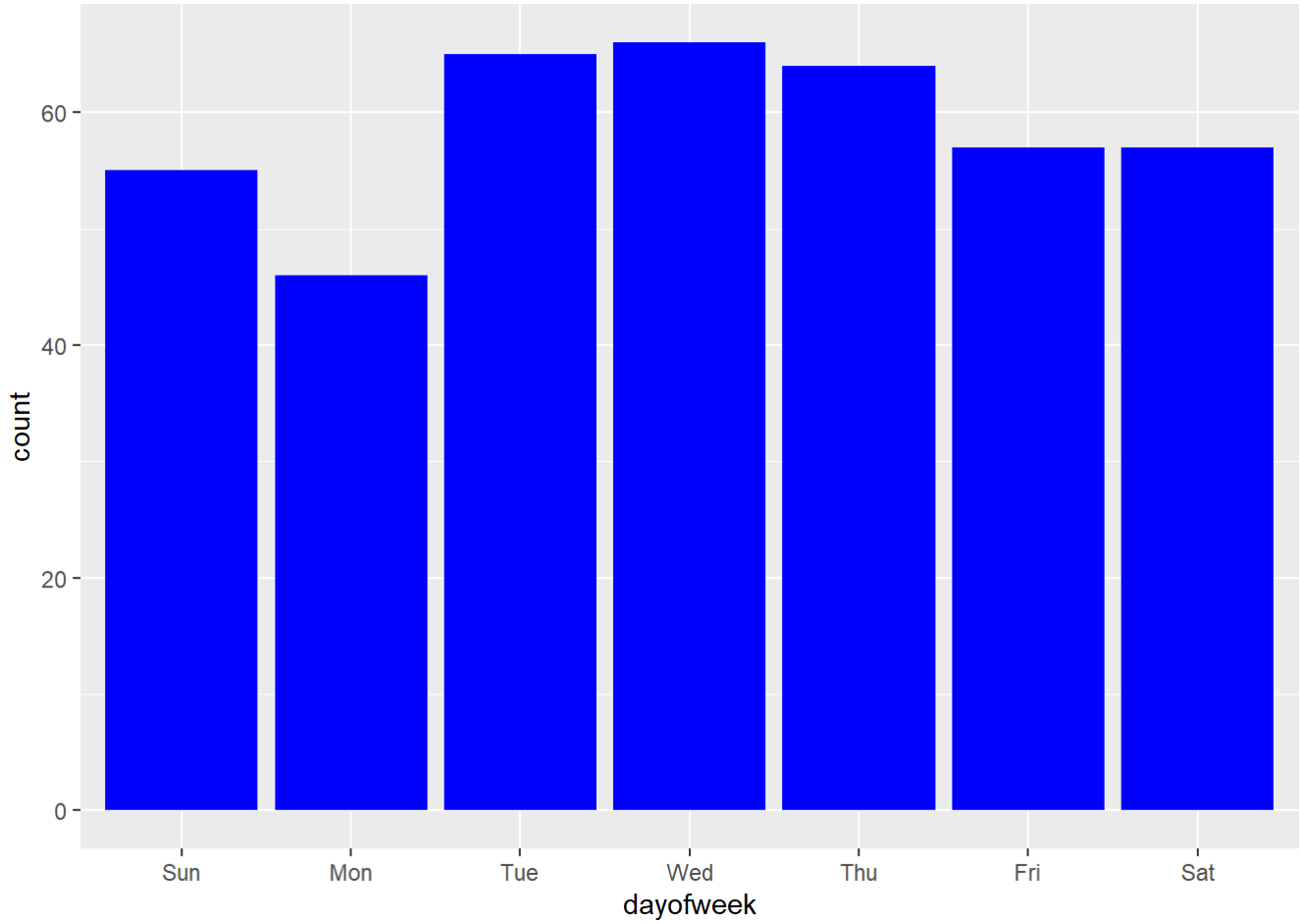
App usage across the week

```
ggplot(daily_activity, aes(x=dayofweek))+
  geom_bar(stat='count', fill = "green")
```



Users use the app more frequently between Tuesday and Thursday. Usage then drops significantly and maintains from Friday to Monday.

```
ggplot(daily_sleep, aes(x=dayofweek))+
  geom_bar(stat='count', fill = "blue")
```

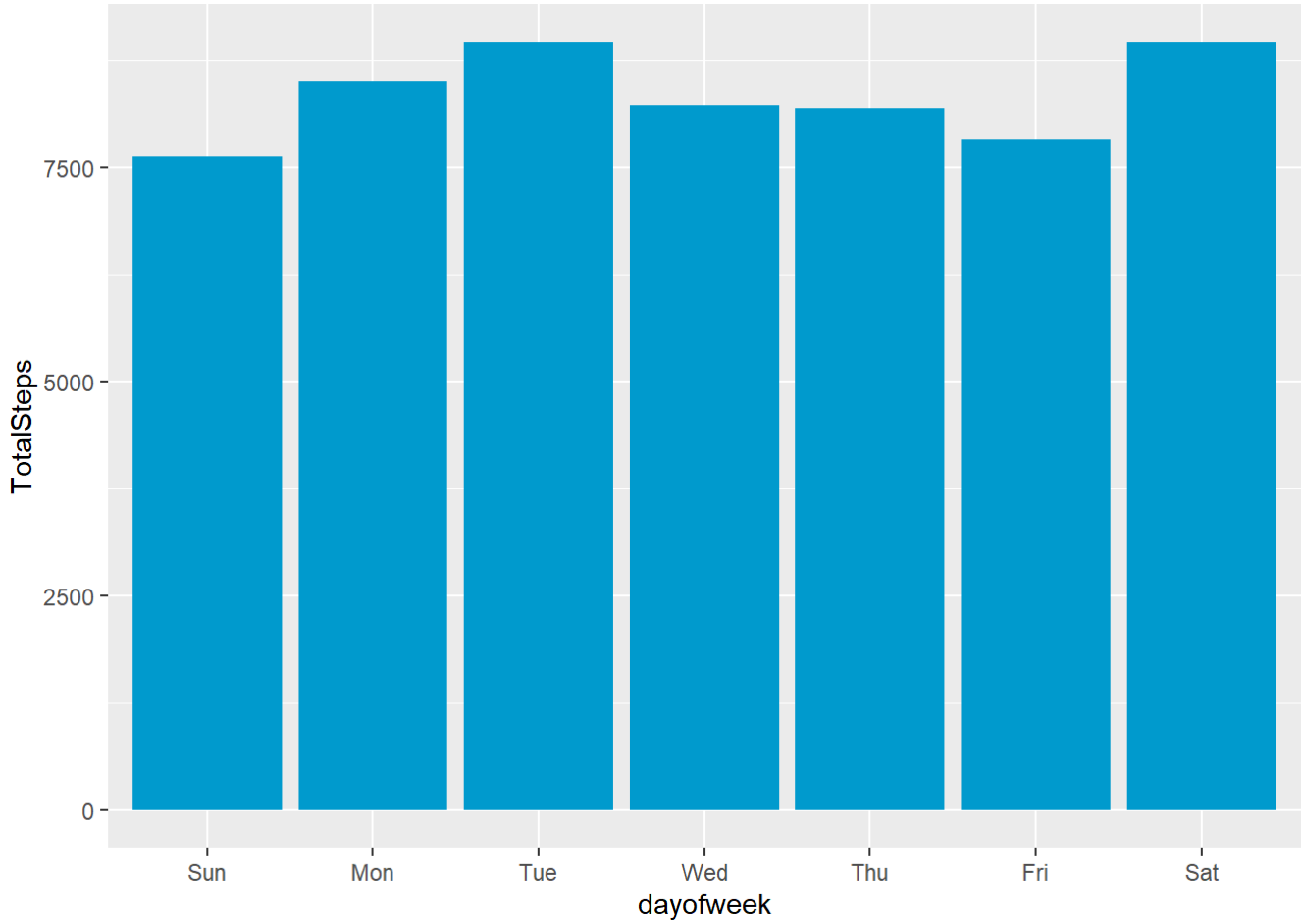


Similarly, users use the app more frequently between Tuesday and Thursday. Usage then drops significantly and maintains from Friday to Sunday. Monday has the lowest usage.

Data did not have any explicit instructions for users to always use the app, which may explain the missing records.

Day of week and Average Steps

```
ggplot(daily_activity, aes(x=dayofweek, y=TotalSteps))+
  geom_bar(stat='summary', fun="mean", fill = "deepskyblue3")
```

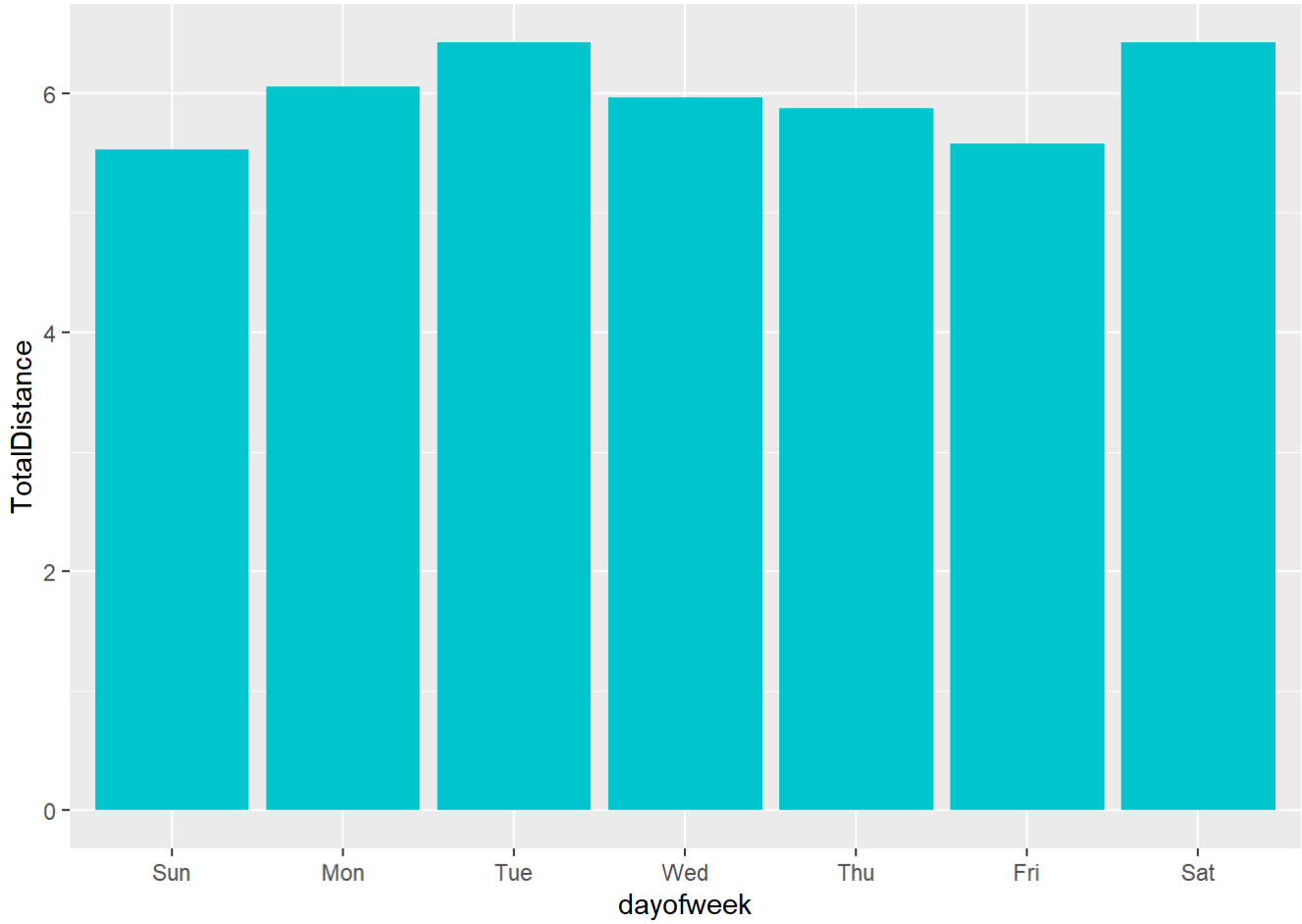


The Centers for Disease Control and Prevention (CDC) recommends that adults should take 10,000 steps per day. (Source: How many steps should people take per day? (https://www.medicalnewstoday.com/articles/how-many-steps-should-you-take-a-day#:~:text=For%20general%20fitness%2C%20most%20adults,Control%20and%20Prevention%20(CDC)%20.))

Participants take the most average number of steps on both Tuesday and Saturday, which hits the recommended number of steps. However, the average number of steps falls below the recommended number on all other days.

Day of week and Average Distance

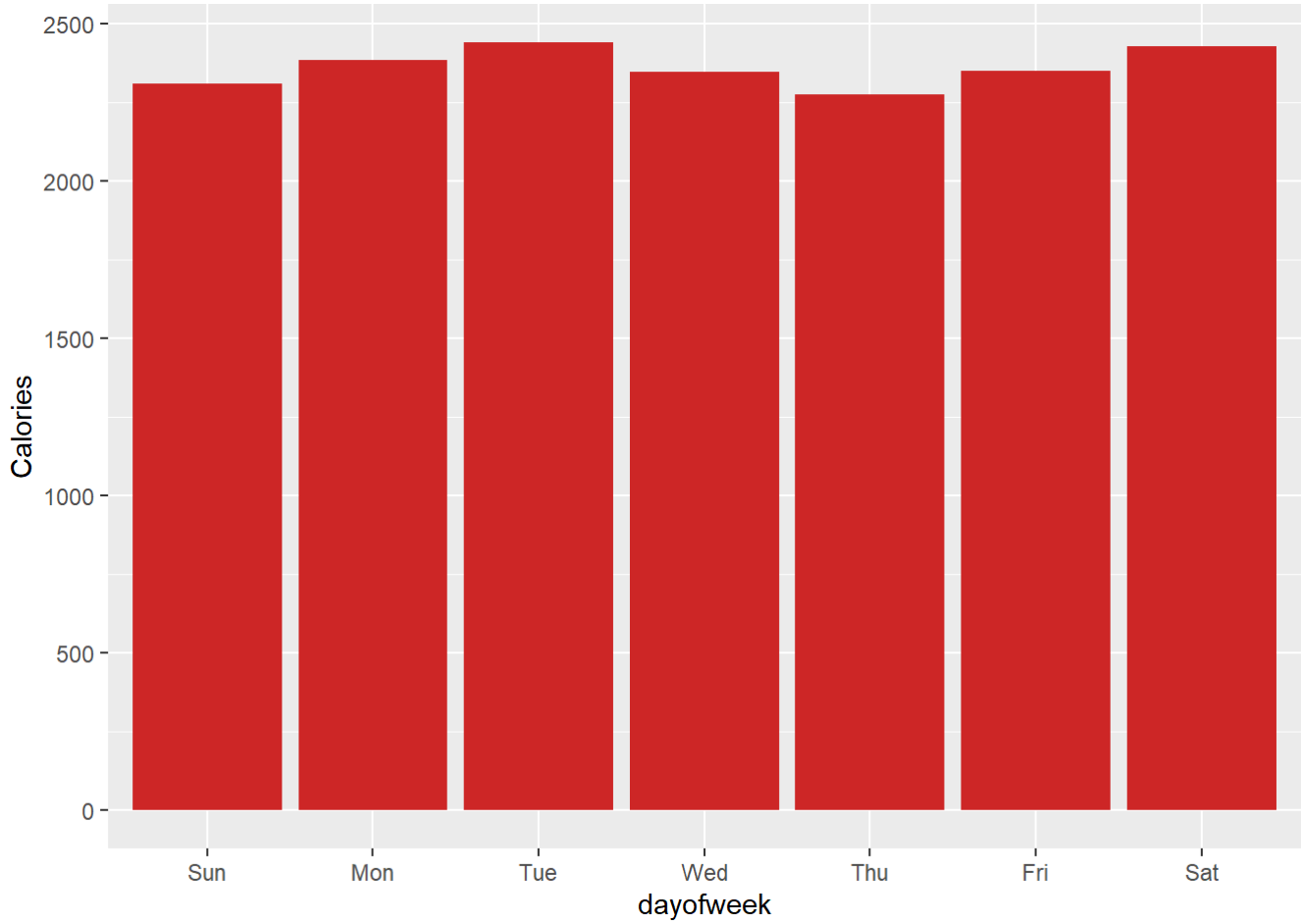
```
ggplot(daily_activity, aes(x=dayofweek, y=TotalDistance))+
  geom_bar(stat='summary',fun = "mean", fill="turquoise3")
```



Greatest average distance traveled is on Saturday while lowest average distance traveled is on Sunday.

Day of week and Average Calories

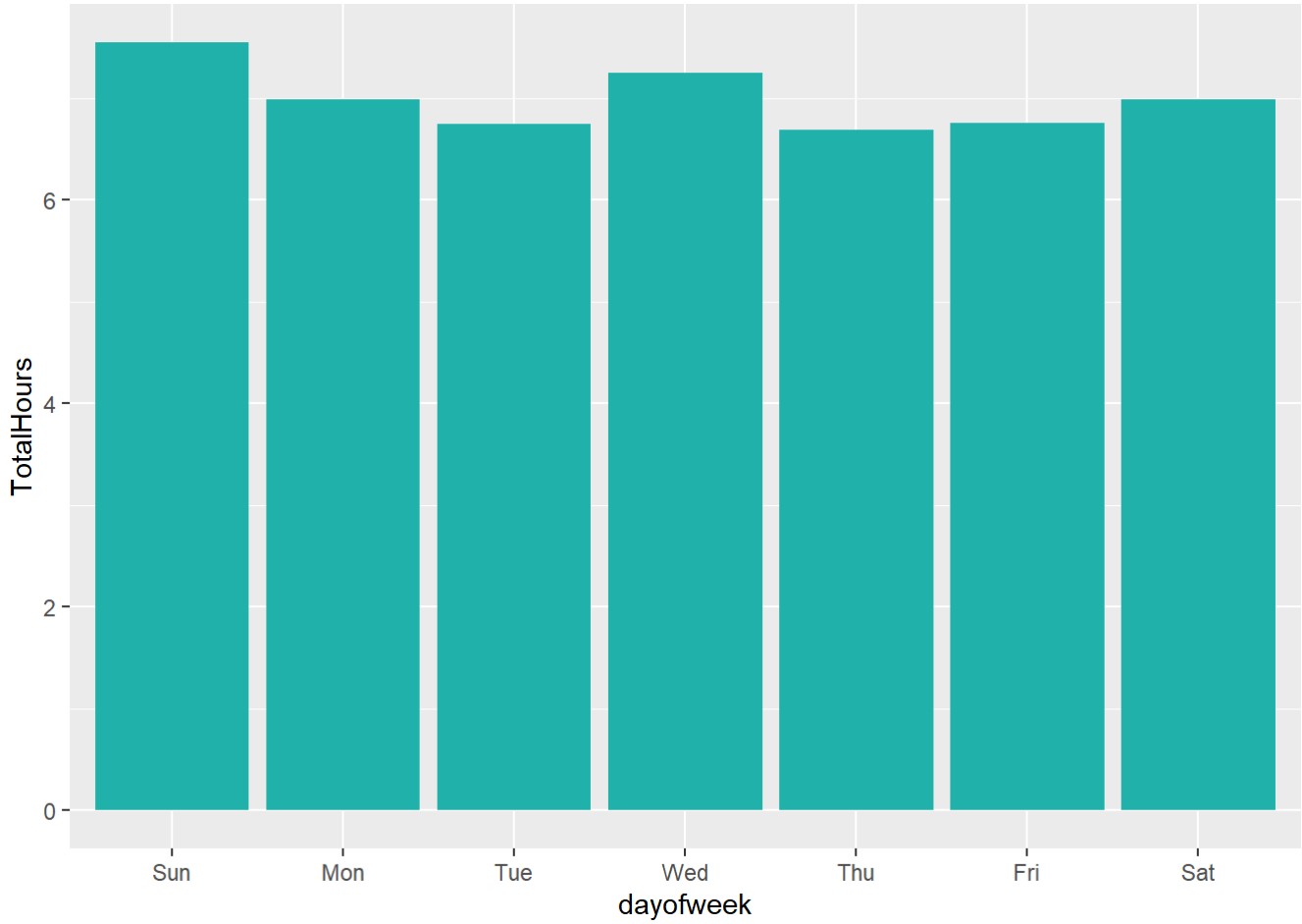
```
ggplot(daily_activity, aes(x=dayofweek, y=Calories))+
  geom_bar(stat='summary', fun ="mean", fill ="firebrick3")
```



Amount of calories burned do not differ too much throughout the week. It is important to note that this data should be used in conjunction with key demographic information such as age, gender, weight etc in order for it to be meaningful. Subsequent analysis could be done with key demographic data.

Day of week and Average Sleep

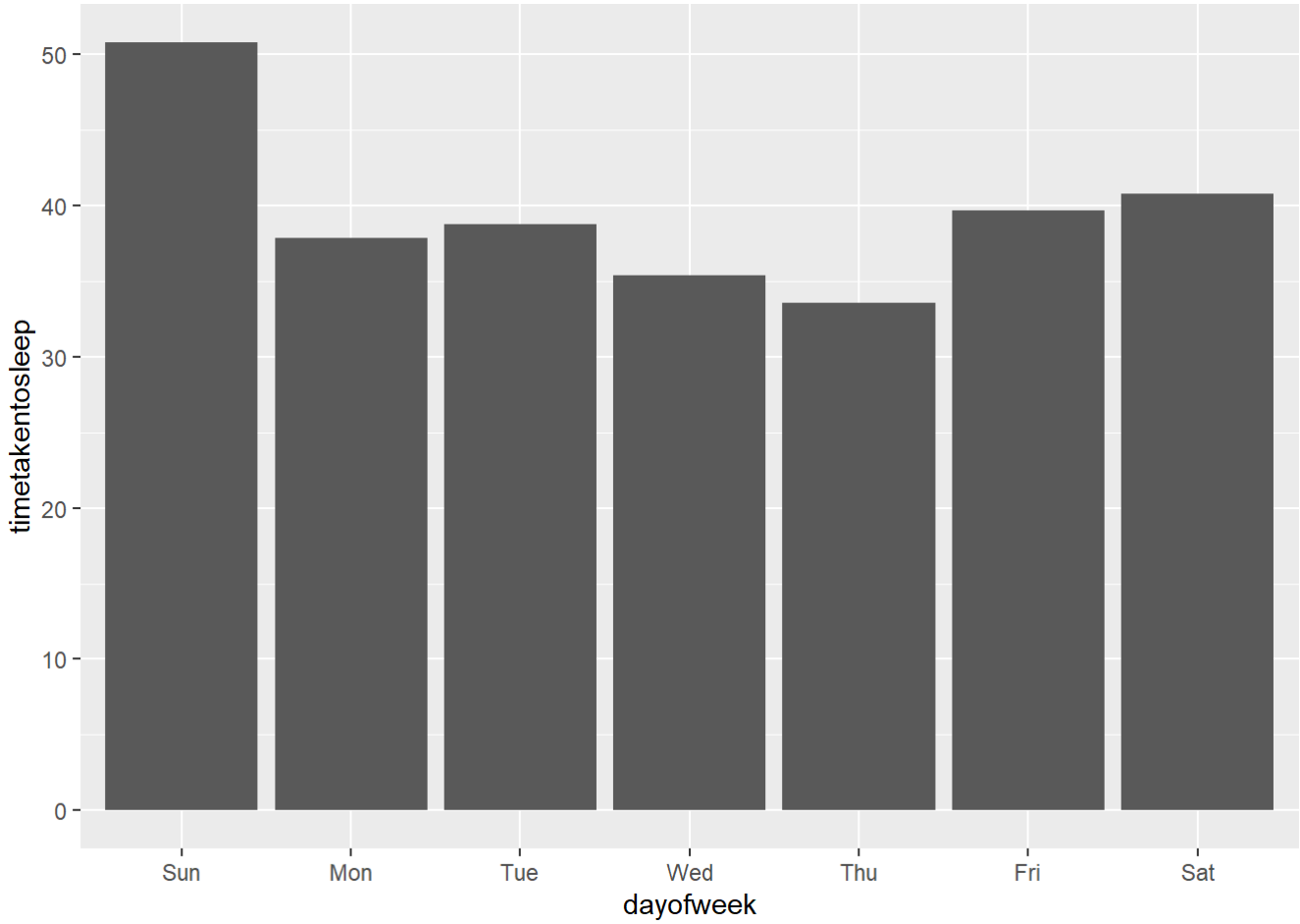
```
ggplot(daily_sleep, aes(x=dayofweek, y=TotalHours))+
  geom_bar(stat='summary', fun="mean", fill = "lightseagreen")
```



Generally, respondents do not obtain the recommended amount of sleep across the week.

Day of Week and time taken to fall asleep

```
ggplot(daily_sleep, aes(x=dayofweek, y=timetakentosleep))+
  geom_bar(stat='summary', fun="mean")
```



Participants take the longest amount of time to fall asleep on Sundays. This could be due to various reasons (e.g. stress from the start of a new work week/inability to adjust body clock due to lifestyle differences on both weekdays and weekends). Further studies could be conducted to investigate further.

Percentage of minutes spent on activity

Investigating the various proportions of minutes spent by respondents.

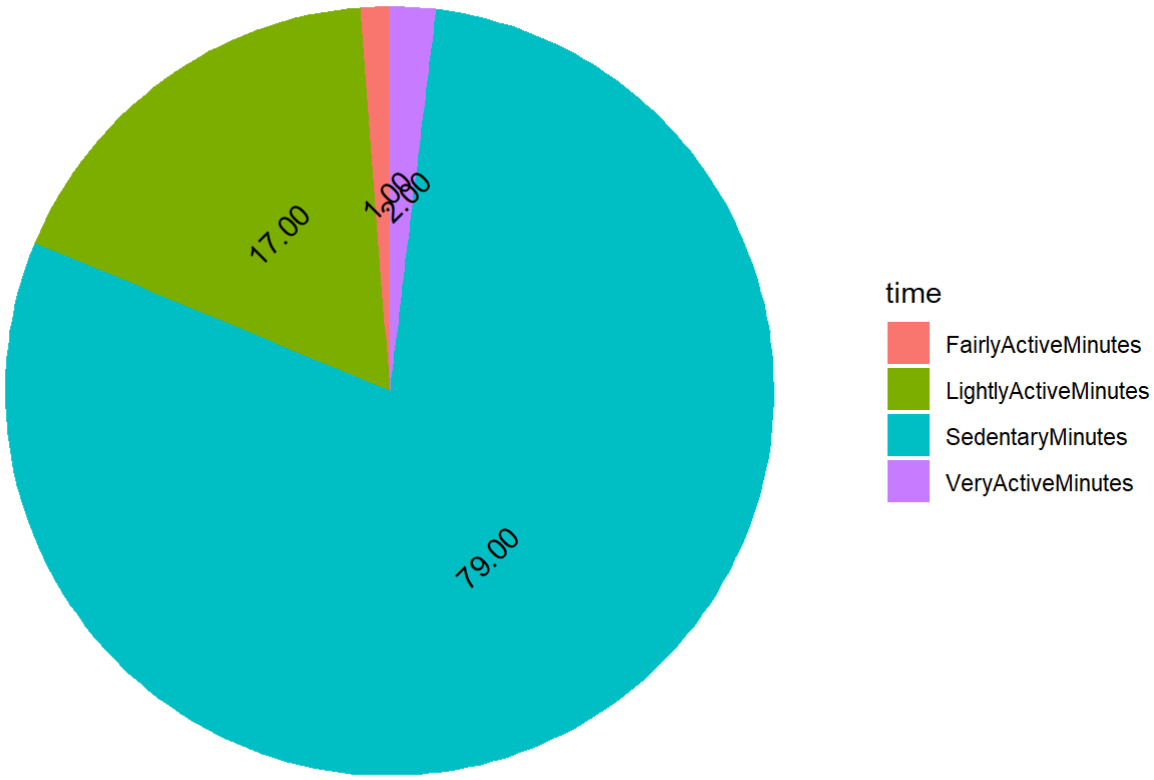
```
## Creating a new dataframe with the required values and headings
minutes.df <- data.frame(time = c("VeryActiveMinutes", "FairlyActiveMinutes", "LightlyActiveM
inutes", "SedentaryMinutes"),
  Value = c(sum(daily_activity$VeryActiveMinutes), sum(daily_activity$FairlyActiveMinutes), s
um(daily_activity$LightlyActiveMinutes), sum(daily_activity$SedentaryMinutes)))


## Calculating the various percentages of minutes
minutes.df <- minutes.df %>%
  mutate(percentage = format(round(Value/sum(minutes.df$Value) *100), nsmall = 2))

minutes.df
```

```
##                   time  Value percentage
## 1    VeryActiveMinutes  19862       2.00
## 2  FairlyActiveMinutes  12751       1.00
## 3 LightlyActiveMinutes 181243      17.00
## 4     SedentaryMinutes 823376      79.00
```

```
ggplot(minutes.df, aes(x="", y=Value, fill=time))+
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start = 0)+
  geom_text(aes(label = percentage, angle = 45), position = position_stack(vjust=0.5))+
  theme_void()
```



Sedentary Minutes take up the largest percentage amongst respondents.

This may suggest that the app is used more as a lifestyle tracker for daily activities such as commuting rather than for fitness.

This can also suggest that users treat functions of the app such as recording number of steps taken, calories burned etc as bonus features and not for their main purpose of monitoring their health and fitness.

Act

For the final step, trends and insights obtained from the data would be used to answer the business task and other relevant questions.

1. What are some trends in smart device usage?

- The data suggests that majority of users utilise the app as a lifestyle tracker for their daily activities (as observed from the largest percentage of sedentary minutes recorded) rather than to monitor their health and fitness.
- There is higher usage on weekdays (particularly from Tuesday to Thursday) than on weekends. This can suggest that using the app is better integrated into their daily routine on weekdays than on weekends, perhaps due to difference in lifestyle on weekdays and weekends.

2. How could these trends apply to Bellabeat customers?

- These trends can be applied to Bellabeat customers by helping Bellabeat develop products that can help women in aspects of health and fitness.
- The products can provide important data surrounding their health and fitness, as well as providing them with the relevant knowledge to help them make better decisions to improve their health and fitness.

3. How could these trends help influence Bellabeat marketing strategy?

- Bellabeat marketing team can adopt the approach of promoting health and fitness by increasing its accessibility and convenience through their products.
- For instance, future products could provide useful tips and information surrounding health and fitness (such as amount of calories burned through a particular exercise)/suggesting new workouts/making health monitoring more interesting and convenient with new functions. These can help to encourage their customers to not only exercise more, but also use more of Bellabeat's products.
- Bellabeat can also promote greater integration of exercising into their customers' daily routine throughout the week using their products.