case_study_2

Author: Joy

Last Updated : 2022-09-07

Project Overview

This whole project is based on a health product related company called Bellbeat , where we will analyze users daily activity, hourly activity and more data to answer some business questions and to provide some recommendations based on our analysis.

The Scenario

Urška Sršen and Sando Mur founded Bellabeat, a high-tech company that manufactures health-focused smart products. Sršen used her background as an artist to develop beautifully designed technology that informs and inspires women around the world. 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.

By 2016, Bellabeat had opened offices around the world and launched multiple products. Bellabeat products became available through a growing number of online retailers in addition to their own e-commerce channel on their website. The company has invested in traditional advertising media, such as radio, out-of-home billboards, print, and television, but focuses on digital marketing extensively. Bellabeat invests year-round in Google Search, maintaining active Facebook and Instagram pages, and consistently engages consumers on Twitter. Additionally, Bellabeat runs video ads on Youtube and display ads on the Google Display Network to support campaigns around key marketing dates.

Business Taks

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?**

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**

Data Source

Our Data Source https://www.kaggle.com/arashnic/fitbit (https://www.kaggle.com/arashnic/fitbit)

Steps

We will follow the steps what we’ve learned so far .

The **Steps are** :

- **Ask**
- **Prepare**
- **Process**
- **Analyze**
- **Share**
- **Act**

Ask

Analyze smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices. To analyse , we have to select one of Bellabeat products . In our case we will analyse data of Fitbit users.

Prepare

- **In this phase we will know about our data.**
- **Then we’ll import the data from source.**
- **After import we will organize those data sets in order to keep everything concise and readable.**

Download the data from here (https://www.kaggle.com/arashnic/fitbit)

After download

- **First create a seperate folder for only this project.**
- **Name the folder .**
- **Move downloaded data sets to the Folder.**
- **Rename data sets to identify clearly**

Importing Data

To import our data sets first we need to install required package to read our csv files.

```
install.packages('tidyverse')
```

After installing the package we will load it up.

```
library(tidyverse)  #Loading the package using library function.
```

Tidyverse is the most popular package to work with data related field . It comes with lots of functionality.

Now we will read those csv files using `read_csv` function included in “Tidyverse” package.

Reading csv files

```
setwd("/Users/joy")
daily_activity = read_csv('Desktop/data_analysis/case_study_2_data/dailyActivity_merged.csv')

#ignore data set
daily_calories = read_csv('Desktop/data_analysis/case_study_2_data/dailyCalories_merged.csv')
daily_intensites = read_csv('Desktop/data_analysis/case_study_2_data/dailyIntensities_merged.csv')
daily_steps = read_csv('Desktop/data_analysis/case_study_2_data/dailySteps_merged.csv')
# end ignore data set

hourly_calories = read_csv('Desktop/data_analysis/case_study_2_data/hourlyCalories_merged.csv')
hourly_intensities = read_csv('Desktop/data_analysis/case_study_2_data/hourlyIntensities_merged.csv')
hourly_steps = read_csv('Desktop/data_analysis/case_study_2_data/hourlySteps_merged.csv')
sleep_day = read_csv('Desktop/data_analysis/case_study_2_data/sleepDay_merged.csv')
weight_log = read_csv('Desktop/data_analysis/case_study_2_data/weightLoginfo_merged.csv')
```

Notice here that we’ve commented ignore data set. That’s because these data sets are redundant. And the data they contain are already inside of daily_activity data set. That’s why we will ignore them

That’s it we are almost done with our “**Prepare**” phase . We have organized our data. Now we will start our “**Process**” part.

Process

In this phase of our data analysis we will clean our data and will add some new fields to work with Analyze part.

We will take a close look to our data sets first

```
glimpse(daily_activity)
```

```
## Rows: 940
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036…
## $ ActivityDate             <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/…
## $ TotalSteps               <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019…
## $ TotalDistance            <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8…
## $ TrackerDistance          <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8…
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5…
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3…
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0…
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ VeryActiveMinutes        <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4…
## $ FairlyActiveMinutes      <dbl> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21…
## $ LightlyActiveMinutes     <dbl> 328, 217, 181, 209, 221, 164, 233, 264, 205, …
## $ SedentaryMinutes         <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818…
## $ Calories                 <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203…
```

There is something unusual , ActivityDate should be type of Date. But we can see that it’s character typed.

Let’s fix it

```
daily_activity$ActivityDate = as.Date(daily_activity$ActivityDate, format='%m/%d/%Y')
```

Check again !

```
glimpse(daily_activity)
```

```
## Rows: 940
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036…
## $ ActivityDate             <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-…
## $ TotalSteps               <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019…
## $ TotalDistance            <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8…
## $ TrackerDistance          <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8…
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5…
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3…
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0…
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ VeryActiveMinutes        <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4…
## $ FairlyActiveMinutes      <dbl> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21…
## $ LightlyActiveMinutes     <dbl> 328, 217, 181, 209, 221, 164, 233, 264, 205, …
## $ SedentaryMinutes         <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818…
## $ Calories                 <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203…
```

There we go

Now let’s check for null values

Daily Activity

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

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

No null values

Hourly Calories

```
sum(is.na(hourly_calories))
```

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

No null values

Hourly Intensities

```
sum(is.na(hourly_intensities))
```

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

No null values

Hourly Steps

```
sum(is.na(hourly_steps))
```

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

No null values

Sleep Day

```
sum(is.na(sleep_day))
```

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

No null values

Weight Log

```
sum(is.na(weight_log))
```

```
## [1] 65
```

Oww , 65 null values let’s figure it why!

```
summary(weight_log$Fat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   22.00   22.75   23.50   23.50   24.25   25.00      65
```

As we can see there is 65 null values in Fat column We would’ve filled this missing values if we were provided age data . Hence we will replace NA values with 0. formula to calculate fat of a person (1.20 x BMI) + (0.23 x Age) - 5.4

```
weight_log = weight_log %>% # Replaced NA values with 0 for column Fat
  replace(is.na(.), 0)
```

Now we will add few new columns for further analysis

Adding day_of_week column to daily_activity

```
daily_activity$day_of_week <- weekdays(as.Date(daily_activity$ActivityDate, format='%
m/%d/%Y'))
```

Now we will add and group data columns in hourly_intensity for analysis

Adding ActivityDate column extracted form ActivityHour

```
hourly_intensities =  hourly_intensities %>%
  mutate(ActivityDate = as.Date(ActivityHour, format='%m/%d/%Y'))
```

Adding WeekDay to track day of the week

```
hourly_intensities$WeekDay = weekdays(hourly_intensities$ActivityDate)
```

Summarizing hourly_intensities data to create a informative visualization

```
hourly_intensities_summary_data = hourly_intensities %>%
  group_by(Id, ActivityDate, WeekDay) %>%
  summarise(Total_Intensity = sum(TotalIntensity)) %>%
  arrange(desc(Total_Intensity))
```

Let’s take a look in summarized data of hourly_intensities

```
head(hourly_intensities_summary_data)
```

```
## # A tibble: 6 × 4
## # Groups:   Id, ActivityDate [6]
##           Id ActivityDate WeekDay  Total_Intensity
##        <dbl> <date>       <chr>              <dbl>
## 1 5577150313 2016-05-01   Sunday               904
## 2 5577150313 2016-04-24   Sunday               901
## 3 5577150313 2016-04-30   Saturday             874
## 4 1624580081 2016-05-01   Sunday               855
## 5 5577150313 2016-04-17   Sunday               833
## 6 5577150313 2016-04-16   Saturday             822
```

Now we will add and group data columns in hourly_steps for further analysis

Adding ActivityDate column extracted form ActivityHour

```
hourly_steps =  hourly_steps %>%
  mutate(ActivityDate = as.Date(ActivityHour, format='%m/%d/%Y'))
```

Adding WeekDay column extracted form ActivityHour

```
hourly_steps$WeekDay = weekdays(hourly_steps$ActivityDate)
```

Summarizing hourly_steps data to create a informative visualization

```
hourly_steps_summary_data = hourly_steps %>%
  group_by(Id, ActivityDate, WeekDay) %>%
  summarise(Total_Steps = sum(StepTotal)) %>%
  arrange(desc(Total_Steps))
```

Let’s take a look in summarized data of hourly_steps

```
head(hourly_steps_summary_data)
```

```
## # A tibble: 6 × 4
## # Groups:   Id, ActivityDate [6]
##           Id ActivityDate WeekDay   Total_Steps
##        <dbl> <date>       <chr>           <dbl>
## 1 1624580081 2016-05-01   Sunday          36019
## 2 8877689391 2016-04-16   Saturday        29204
## 3 8877689391 2016-04-30   Saturday        27710
## 4 8877689391 2016-04-27   Wednesday       23603
## 5 8877689391 2016-04-12   Tuesday         23186
## 6 8053475328 2016-04-24   Sunday          22988
```

There we go , almost done with our process phase. Now we will Analyze our data and point out key insights

Analyze

In this part of analysis we will discover and find some key findings to answer our business question and to provide recommendations.

First we will view the summary of all data sets

Daily Activity

```
summary(daily_activity)
```

```
##        Id             ActivityDate          TotalSteps    TotalDistance
##  Min.   :1.504e+09   Min.   :2016-04-12   Min.   :    0   Min.   : 0.000
##  1st Qu.:2.320e+09   1st Qu.:2016-04-19   1st Qu.: 3790   1st Qu.: 2.620
##  Median :4.445e+09   Median :2016-04-26   Median : 7406   Median : 5.245
##  Mean   :4.855e+09   Mean   :2016-04-26   Mean   : 7638   Mean   : 5.490
##  3rd Qu.:6.962e+09   3rd Qu.:2016-05-04   3rd Qu.:10727   3rd Qu.: 7.713
##  Max.   :8.878e+09   Max.   :2016-05-12   Max.   :36019   Max.   :28.030
##  TrackerDistance  LoggedActivitiesDistance VeryActiveDistance
##  Min.   : 0.000   Min.   :0.0000           Min.   : 0.000
##  1st Qu.: 2.620   1st Qu.:0.0000           1st Qu.: 0.000
##  Median : 5.245   Median :0.0000           Median : 0.210
##  Mean   : 5.475   Mean   :0.1082           Mean   : 1.503
##  3rd Qu.: 7.710   3rd Qu.:0.0000           3rd Qu.: 2.053
##  Max.   :28.030   Max.   :4.9421           Max.   :21.920
##  ModeratelyActiveDistance LightActiveDistance SedentaryActiveDistance
##  Min.   :0.0000           Min.   : 0.000      Min.   :0.000000
##  1st Qu.:0.0000           1st Qu.: 1.945      1st Qu.:0.000000
##  Median :0.2400           Median : 3.365      Median :0.000000
##  Mean   :0.5675           Mean   : 3.341      Mean   :0.001606
##  3rd Qu.:0.8000           3rd Qu.: 4.782      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.:127.0        1st Qu.: 729.8
##  Median :  4.00    Median :  6.00      Median :199.0        Median :1057.5
##  Mean   : 21.16    Mean   : 13.56      Mean   :192.8        Mean   : 991.2
##  3rd Qu.: 32.00    3rd Qu.: 19.00      3rd Qu.:264.0        3rd Qu.:1229.5
##  Max.   :210.00    Max.   :143.00      Max.   :518.0        Max.   :1440.0
##     Calories    day_of_week
##  Min.   :   0   Length:940
##  1st Qu.:1828   Class :character
##  Median :2134   Mode  :character
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900
```

Daily activity summary takeaway

This summary shows the average user is taking `7638` steps a day, missing the recommended `10,000` steps for health by the “CDC”. On average, users are getting `21.16` minutes of very active or vigorous activity a day, this equates to `148.12` minutes a week. The “CDC” recommends `75` minutes of vigorous activity a week, so the typical Fitbit user is doing well in this area and achieving additional health benefits. In contrast, participants are averaging `991.2` minutes, or `16.52` hours of sedentary time a day! This is a significant amount of time and can lead to other health issues because the body functions best upright. Scientists have determined that 40 minutes of moderate to vigorous activity a day will balance out the effects of sitting up to 10 hours a day. Furthermore, this summary shows the average user is burning `2304` calories a day. Studies show the average person in the population burns `1800` calories a day, but burning `3500` is needed to lose a pound of weight. The Fitbit users in this case are burning more than the norm, and are on track to lose a few pounds a week if they so choose to.

Hourly Calories

```
summary(hourly_calories)
```

```
##        Id             ActivityHour          Calories     
##  Min.   :1.504e+09   Length:22099       Min.   : 42.00  
##  1st Qu.:2.320e+09   Class :character   1st Qu.: 63.00  
##  Median :4.445e+09   Mode  :character   Median : 83.00  
##  Mean   :4.848e+09                      Mean   : 97.39  
##  3rd Qu.:6.962e+09                      3rd Qu.:108.00  
##  Max.   :8.878e+09                      Max.   :948.00  
```

Hourly calories summary takeaways

From the summary we can see that on a average scale a person is burning up to `97.39` calories per hour Which is `97.39*24 = 2337.36` if we take 24 hours on scale. And the maximum value is `948` , we know that burning `3500` calories is needed to lose a “pound” , i think those who are burning at this rate are in hurry to lose all their extra “weights” :)

Hourly Intensity

```
summary(hourly_intensities_summary_data)
```

```
##        Id              ActivityDate          WeekDay          Total_Intensity
##  Min.   :1.504e+09   Min.   :2016-04-12   Length:934         Min.   :  0.0  
##  1st Qu.:2.320e+09   1st Qu.:2016-04-19   Class :character   1st Qu.:164.0  
##  Median :4.445e+09   Median :2016-04-26   Mode  :character   Median :300.0  
##  Mean   :4.847e+09   Mean   :2016-04-26                      Mean   :284.8  
##  3rd Qu.:6.962e+09   3rd Qu.:2016-05-04                      3rd Qu.:398.0  
##  Max.   :8.878e+09   Max.   :2016-05-12                      Max.   :904.0  
```

Hourly Intensity summary takeaways

From our summary we can see that a person on average scale is in intense activity of `12.04` minutes per hour, which is 1 `2.04*24 = 288.96 minutes` equivalent to `288.96/60 = 4.816` or around `5 hours` of intense activity each day And the highest value intensity of a single is `904.0` which is almost `904/60 = 15.06 hour` on Sunday.

Sleep Day

```
summary(sleep_day)
```

```
##        Id              SleepDay         TotalSleepRecords TotalMinutesAsleep
##  Min.   :1.504e+09   Length:413         Min.   :1.000     Min.   : 58.0
##  1st Qu.:3.977e+09   Class :character   1st Qu.:1.000     1st Qu.:361.0
##  Median :4.703e+09   Mode  :character   Median :1.000     Median :433.0
##  Mean   :5.001e+09                      Mean   :1.119     Mean   :419.5
##  3rd Qu.:6.962e+09                      3rd Qu.:1.000     3rd Qu.:490.0
##  Max.   :8.792e+09                      Max.   :3.000     Max.   :796.0
##  TotalTimeInBed
##  Min.   : 61.0
##  1st Qu.:403.0
##  Median :463.0
##  Mean   :458.6
##  3rd Qu.:526.0
##  Max.   :961.0
```

Sleep Day summary takeaways

The summary of the sleep data frame displays the average user sleeps once per day for `419.5` minutes, or roughly `7 hours`. This falls within the “CDC’s” recommendations for adults in order to get the proper amount of rest. The average participant is spending `458.6 minutes` in bed, or `7.64 hours`. This means the typical user is spending `38.6 minutes` awake in bed. According to Health Central, people should not spend more than 1 hour in bed awake. This is to prevent a mental link being formed between being awake and being in bed, which can lead to insomnia.

Weight Log

```
summary(weight_log)
```

```
##        Id                Date              WeightKg       WeightPounds
##  Min.   :1.504e+09   Length:67          Min.   : 52.60   Min.   :116.0
##  1st Qu.:6.962e+09   Class :character   1st Qu.: 61.40   1st Qu.:135.4
##  Median :6.962e+09   Mode  :character   Median : 62.50   Median :137.8
##  Mean   :7.009e+09                      Mean   : 72.04   Mean   :158.8
##  3rd Qu.:8.878e+09                      3rd Qu.: 85.05   3rd Qu.:187.5
##  Max.   :8.878e+09                      Max.   :133.50   Max.   :294.3
##       Fat               BMI        IsManualReport      LogId
##  Min.   : 0.0000   Min.   :21.45   Mode :logical   Min.   :1.460e+12
##  1st Qu.: 0.0000   1st Qu.:23.96   FALSE:26        1st Qu.:1.461e+12
##  Median : 0.0000   Median :24.39   TRUE :41        Median :1.462e+12
##  Mean   : 0.7015   Mean   :25.19                   Mean   :1.462e+12
##  3rd Qu.: 0.0000   3rd Qu.:25.56                   3rd Qu.:1.462e+12
##  Max.   :25.0000   Max.   :47.54                   Max.   :1.463e+12
```

Weight Log summary takeaways

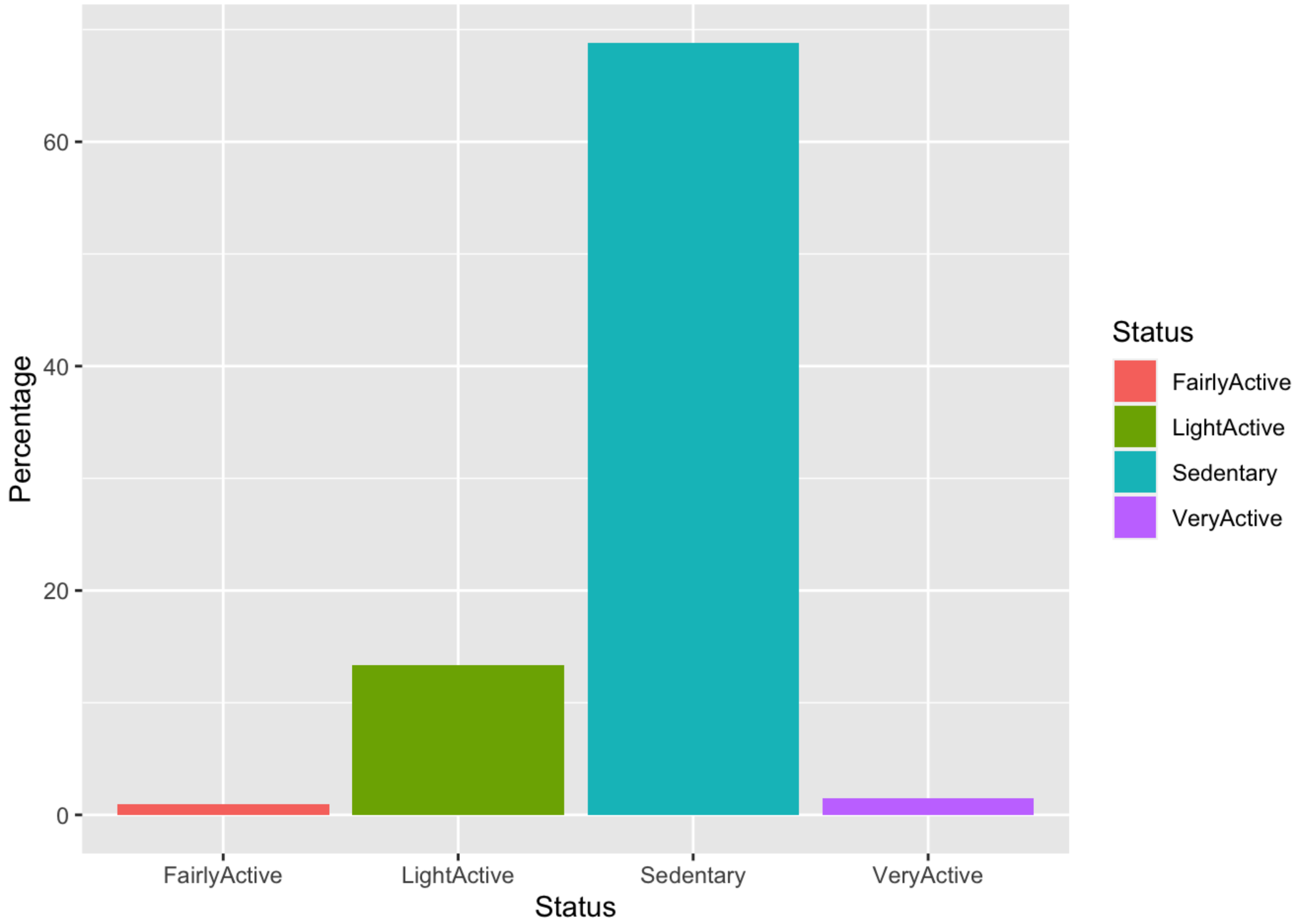
Data frame has a low number of participants, and the average BMI is `25.19`. This is considered as overweight BMI. However, BMI can be a screening tool and does not diagnose the body fatness or health of an person.

That’s it we’re almost done with Analyze part. Now we will visualize what we have found while Analyzing

Share

Daily Activity overview

```
min_in_day = 60*24
Sedentary = (mean(daily_activity$SedentaryMinutes)/ min_in_day ) * 100
VeryActive = (mean(daily_activity$VeryActiveMinutes)/ min_in_day) * 100
LightActive = (mean(daily_activity$LightlyActiveMinutes)/ min_in_day) * 100
FairlyActive = (mean(daily_activity$FairlyActiveMinutes)/ min_in_day) * 100

activity_df = data.frame(Status = c('Sedentary', 'VeryActive', 'LightActive', 'Fairly
Active'),
                         Percentage = c(Sedentary, VeryActive, LightActive, FairlyAct
ive))


ggplot(activity_df) + geom_col(mapping=aes(x=Status, y=Percentage, fill=Status))
```



From the visualization we can see that majority of users almost `( 69% )` percentage are spending time in sedentary state . And only `13%` of them are Lightly Active.

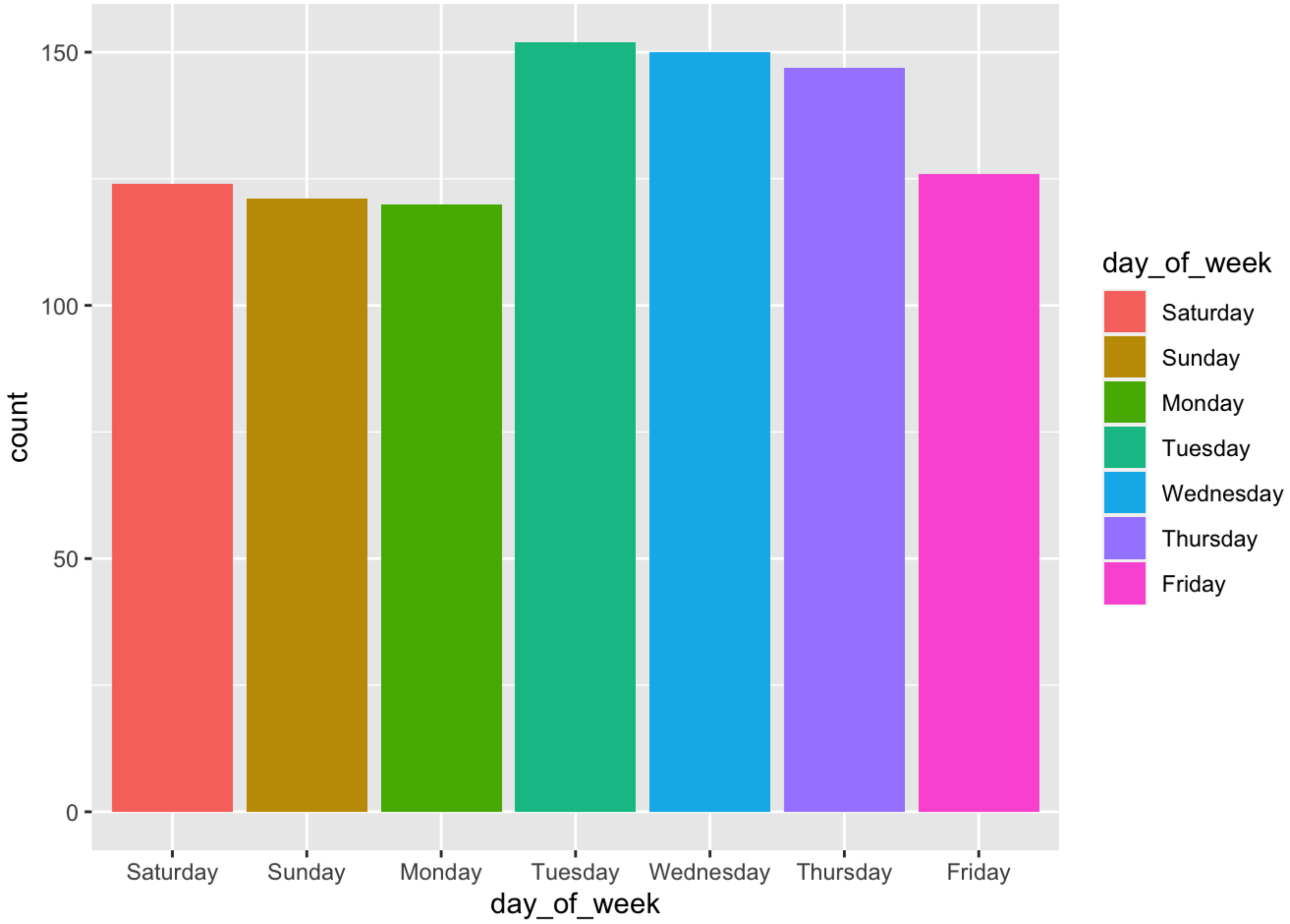
Active weekday frequency

Formatting day_of_week for clear visualization

```
daily_activity$day_of_week <- factor(daily_activity$day_of_week, levels=c('Saturday',
'Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'))
```

Frequency

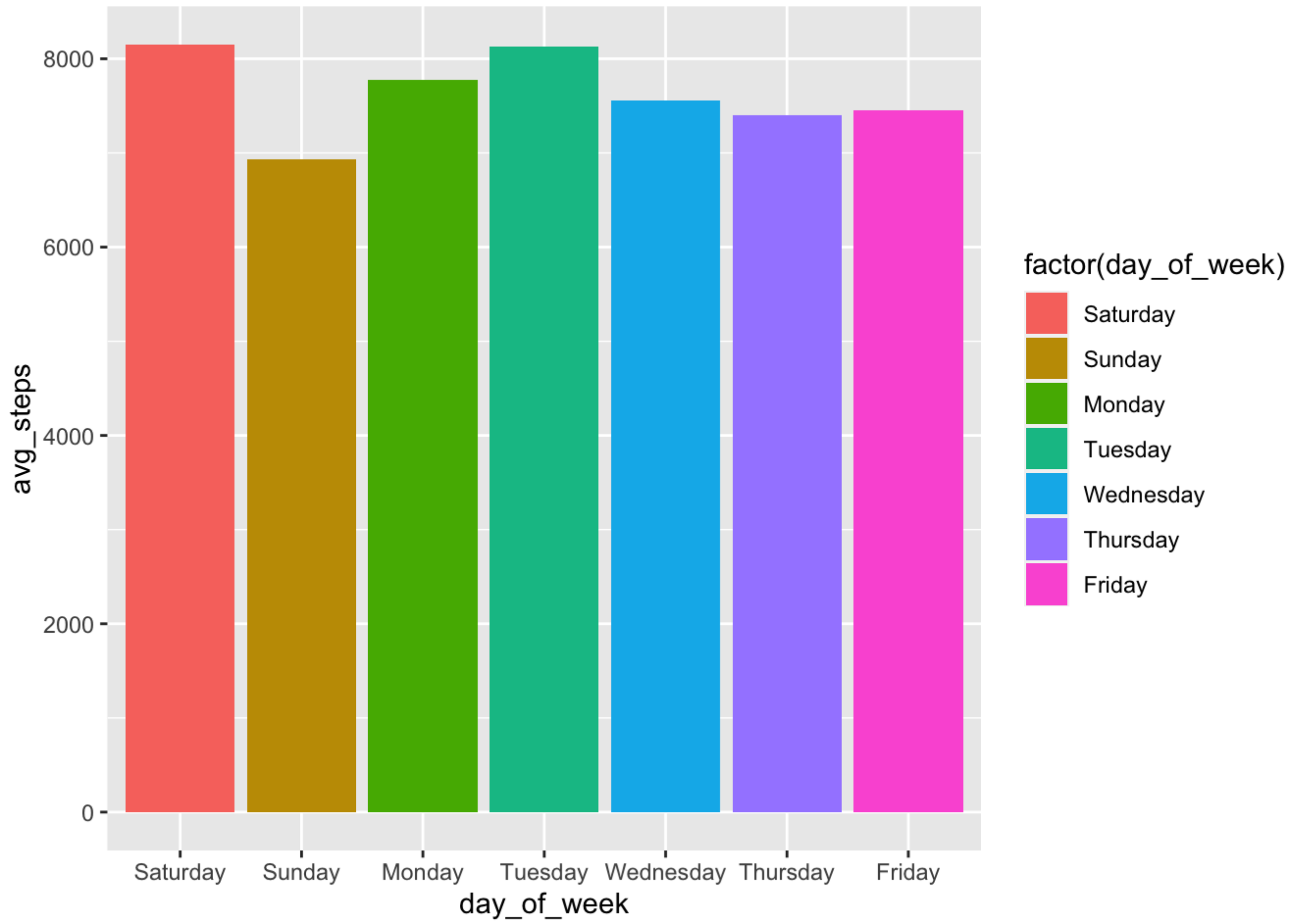
```
ggplot(daily_activity) + geom_bar(mapping=aes(x=day_of_week, fill=day_of_week))
```



From the above visualization we can see that users are most active on “Tuesday” , “Wednesday” “Thursday” and least active on “Sunday”, “Monday”.

Steps weekday frequency

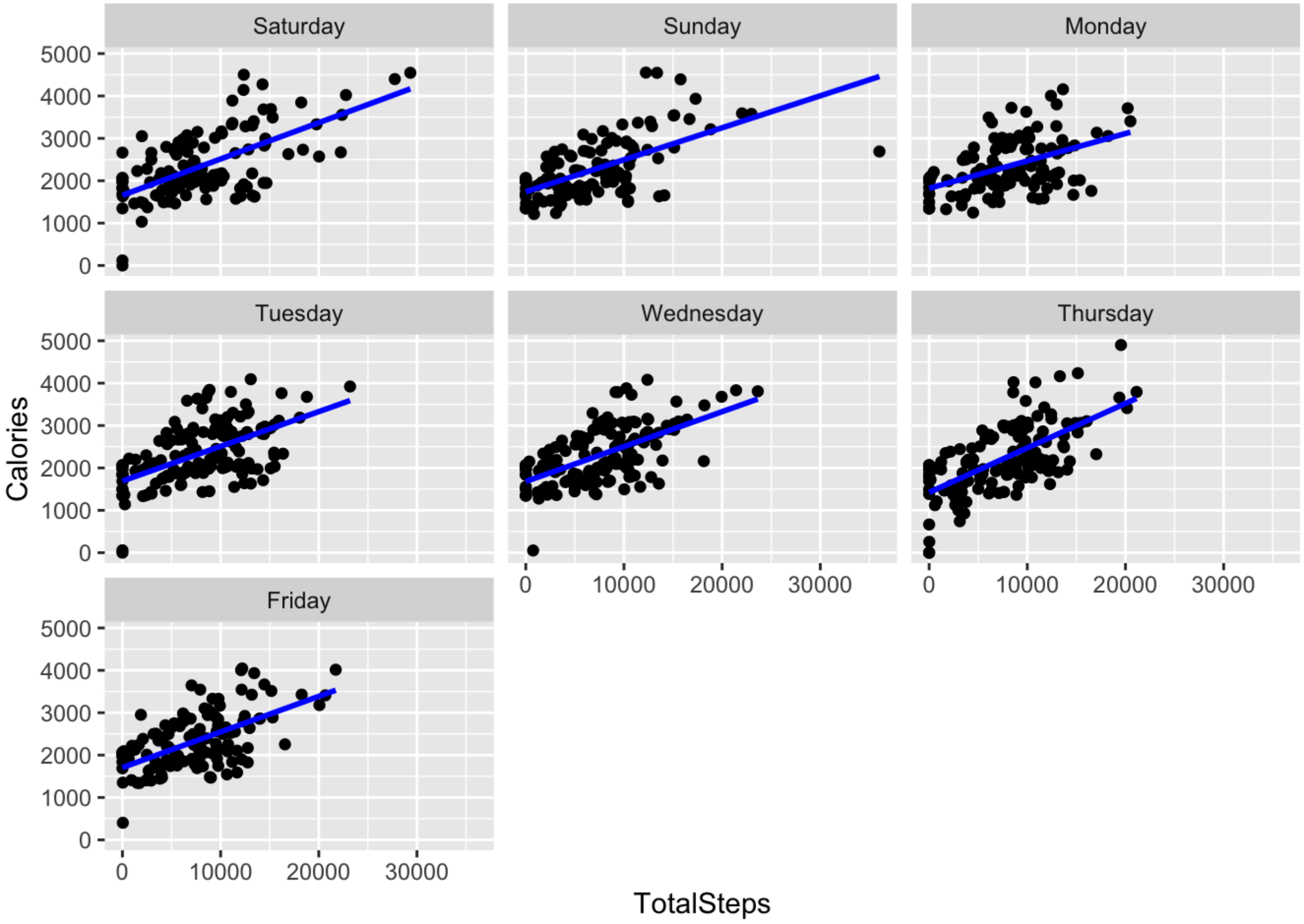
```
step_freq = daily_activity %>%
  group_by(day_of_week) %>%
  summarise(avg_steps = mean(TotalSteps))

ggplot(step_freq) + geom_col(mapping=aes(x=day_of_week, y=avg_steps, fill=factor(day_
of_week)))
```



From the above summary we can see that on average scale users didn’t achieved recommended `10000` steps in any day of the week. But the highest average steps are from Saturday and Tuesday. And the Lowest is from Sunday , we can conclude that users are being lazy at weekends.

Steps vs Calories

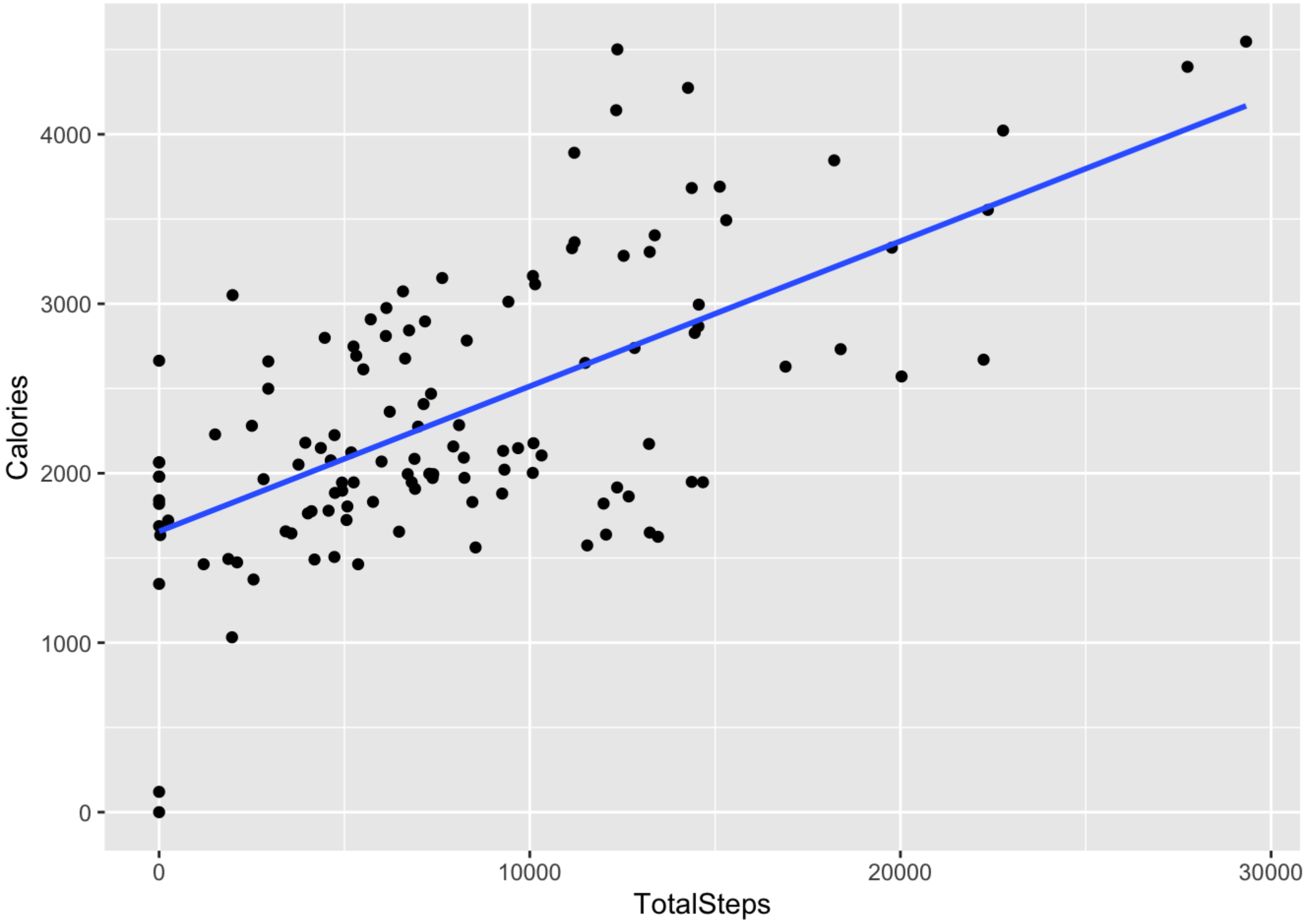
```
ggplot(daily_activity, aes(x=TotalSteps, y=Calories)) + geom_point() +
  geom_smooth(col='blue', method=lm, se=FALSE) +
  facet_wrap(~day_of_week)
```



This visualization is too much faded , lets try to separate these visualizations by week day.

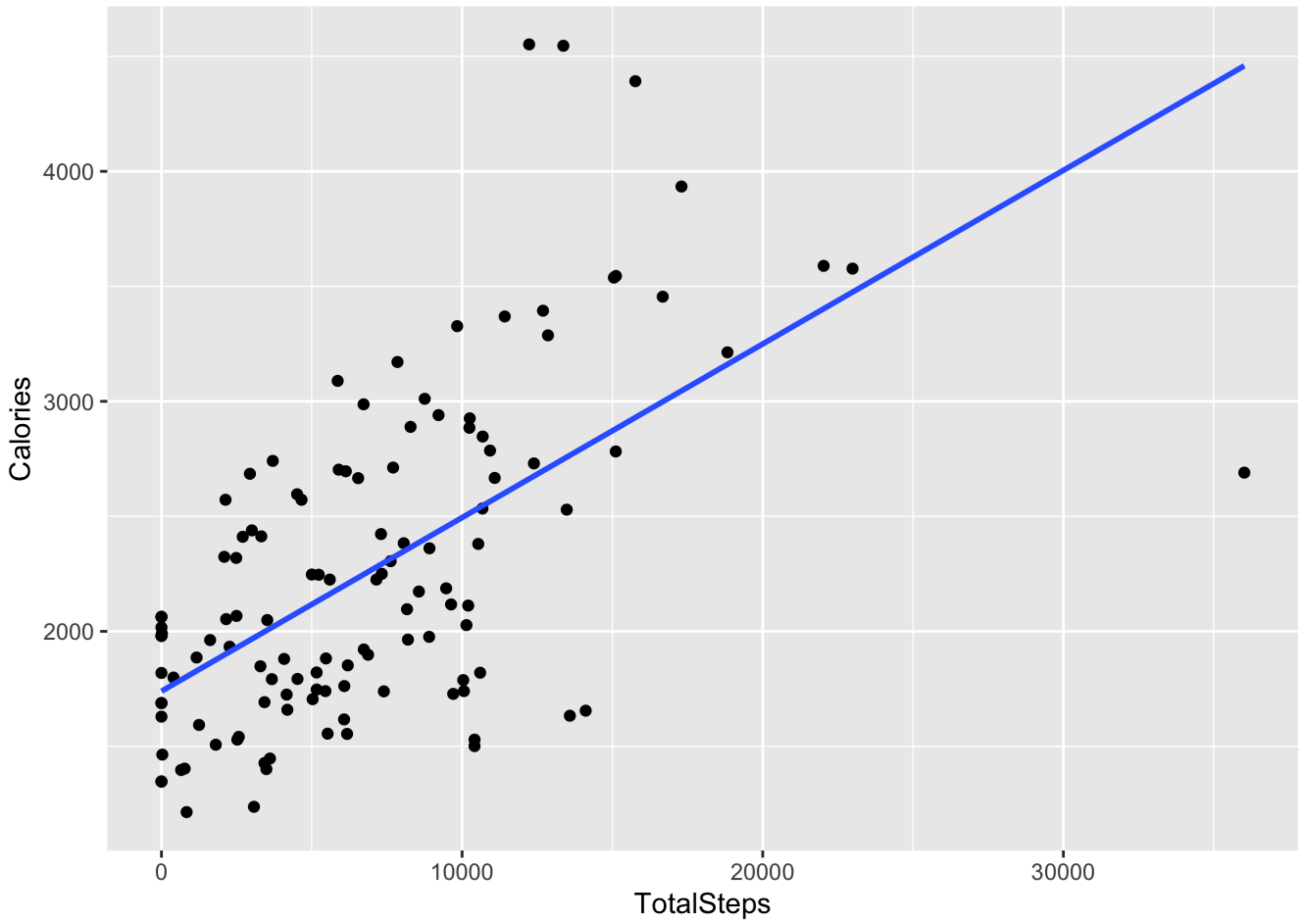
Saturday

```
ggplot(filter(daily_activity, day_of_week=='Saturday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



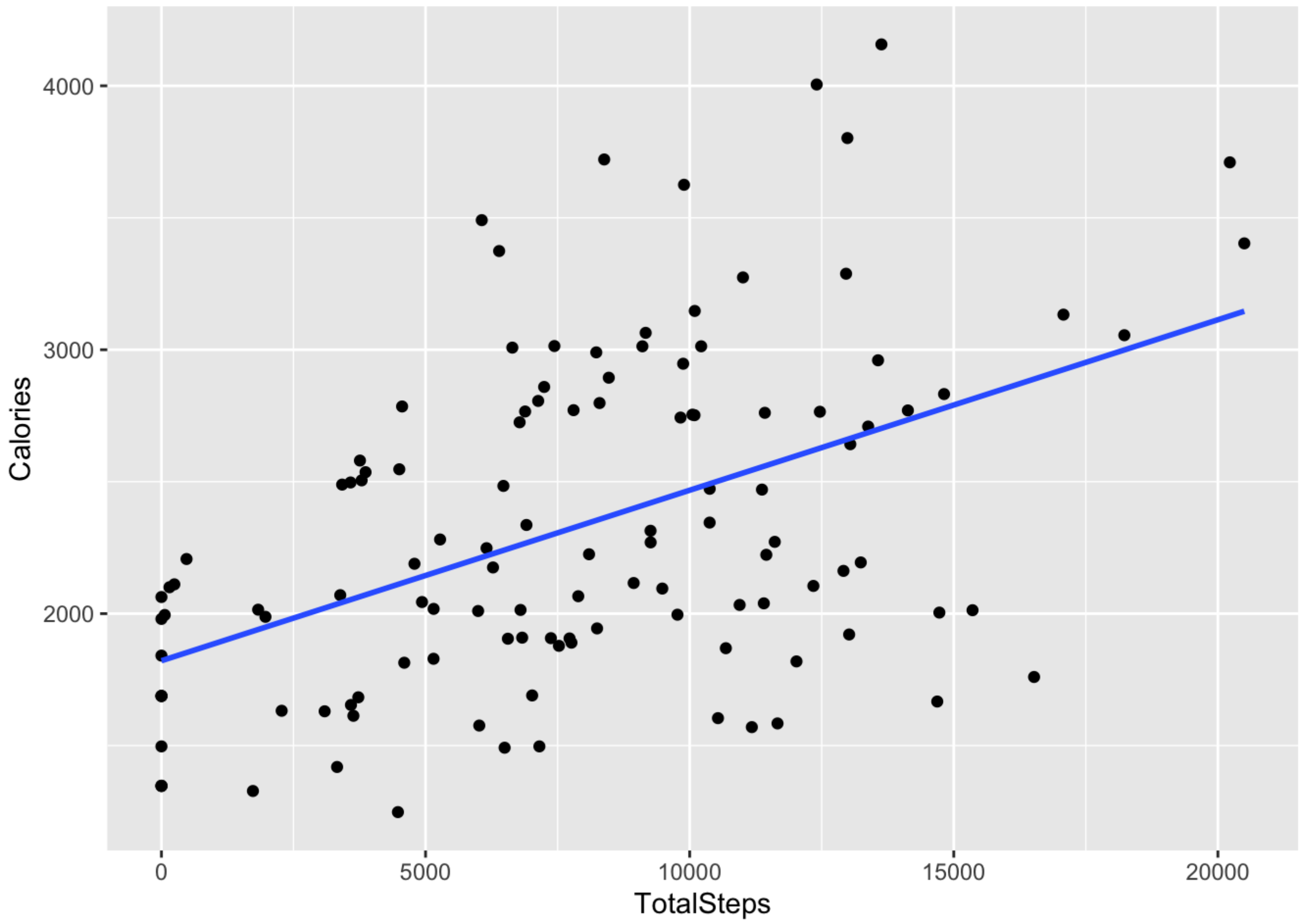
Sunday

```
ggplot(filter(daily_activity, day_of_week=='Sunday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



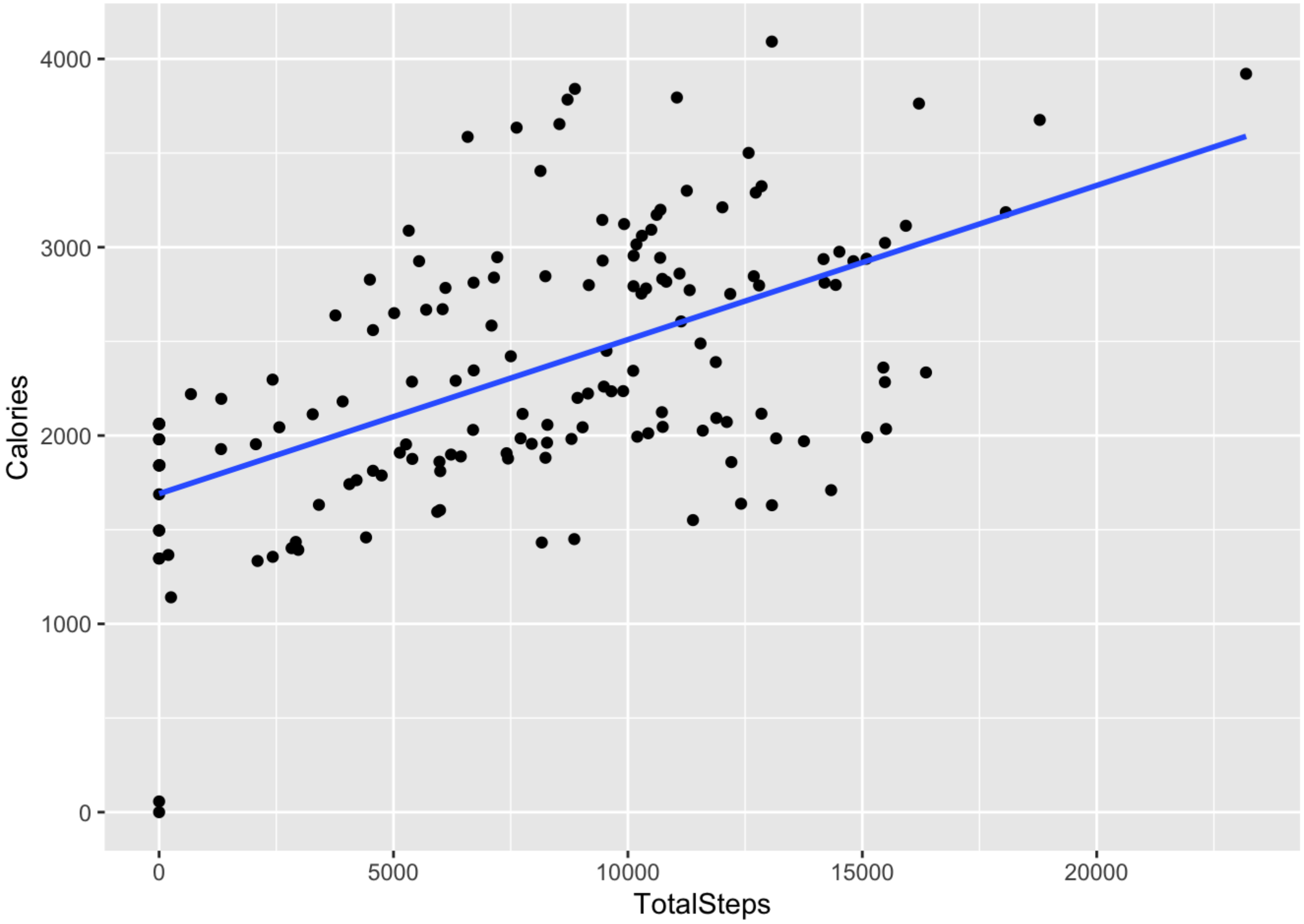
Monday

```
ggplot(filter(daily_activity, day_of_week=='Monday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



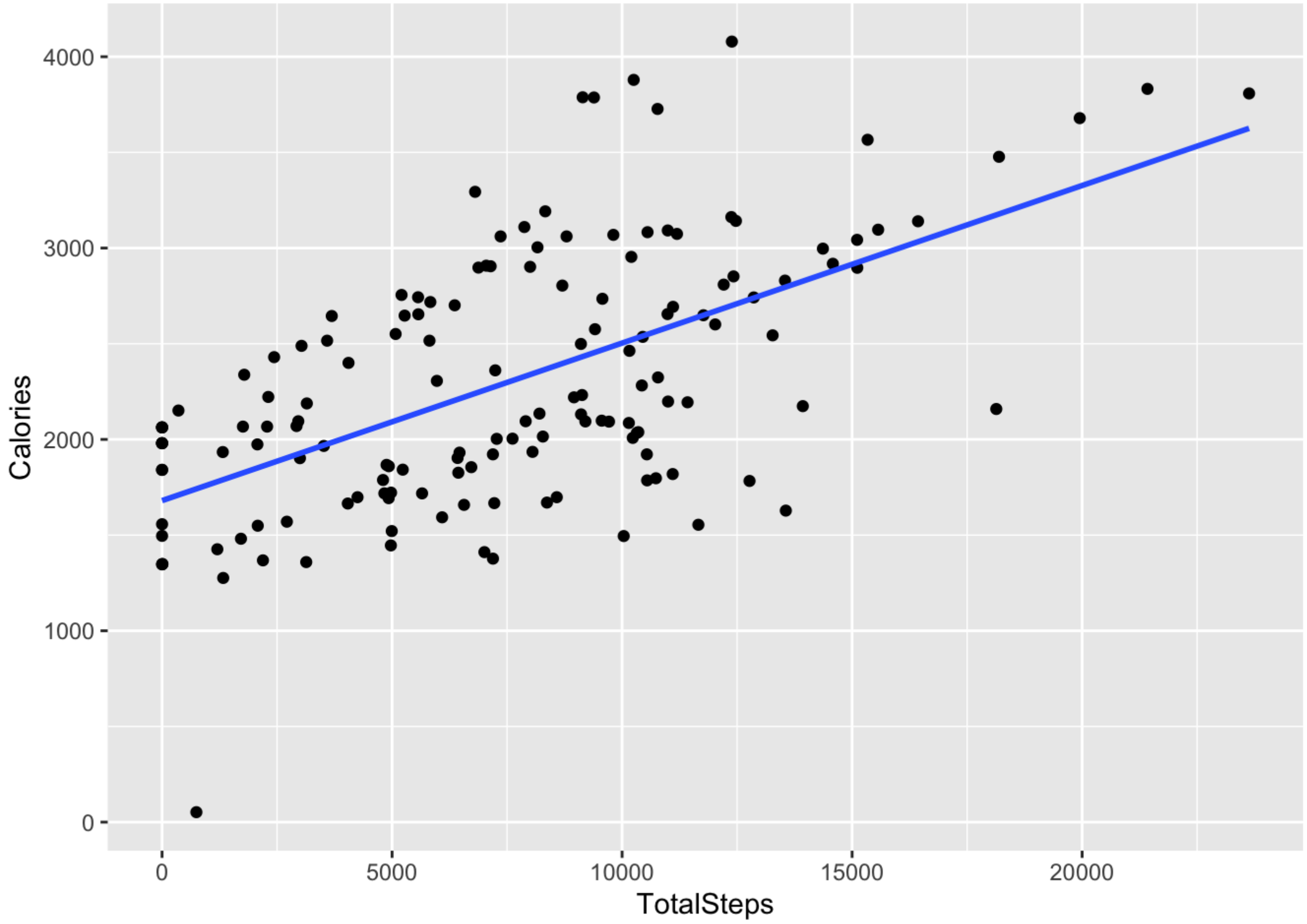
Tuesday

```
ggplot(filter(daily_activity, day_of_week=='Tuesday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



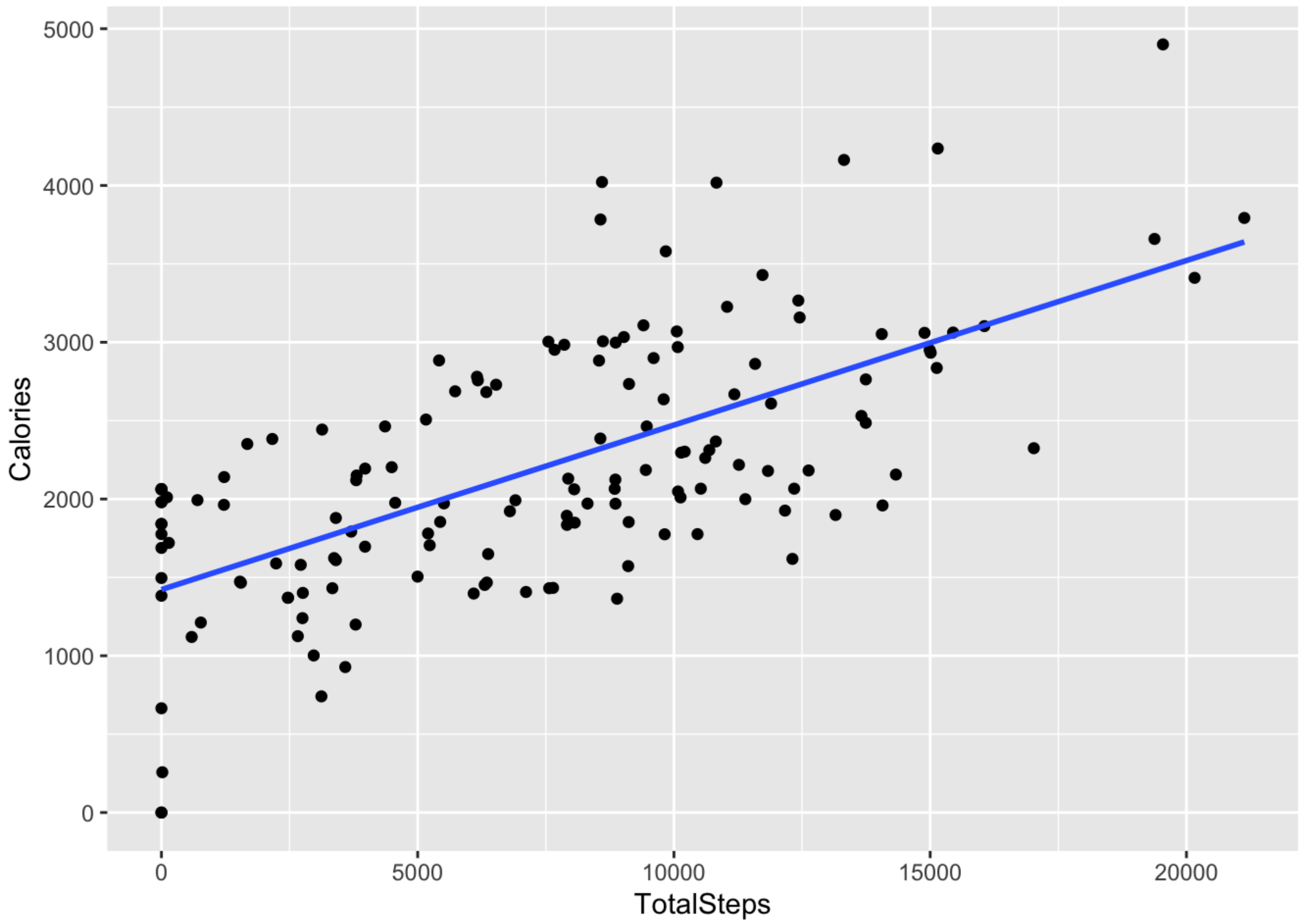
Wednesday

```
ggplot(filter(daily_activity, day_of_week=='Wednesday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



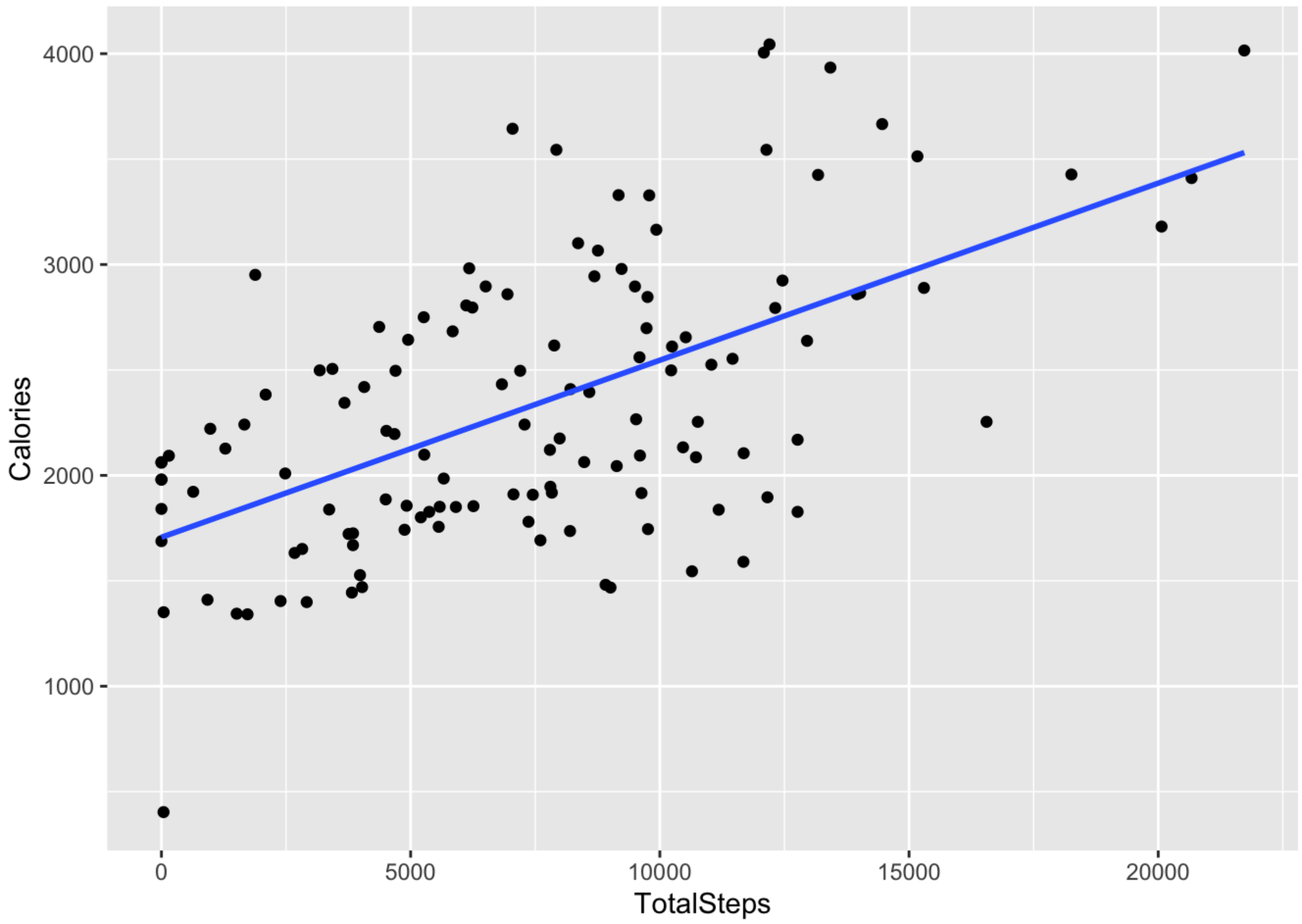
Thursday

```
ggplot(filter(daily_activity, day_of_week=='Thursday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



Friday

```
ggplot(filter(daily_activity, day_of_week=='Friday'), aes(TotalSteps, Calories)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE)
```



From above all scatter plots , we can see that their is a positive relationship between Steps and Calories burned. The more you walk , more you will burn Calories.

Sleep vs Calories

First we need to merge `daily_activity` with `sleep_data` in order to get `Calories` data.

Adding ActivityDate column extracted form SleepDay Column

```
sleep_day =  sleep_day %>%
  mutate(ActivityDate = as.Date(SleepDay, format='%m/%d/%Y'))
```

Adding `day_of_week` column to `sleep_day`

```
sleep_day$day_of_week = weekdays(sleep_day$ActivityDate)
```

Selecting only few columns from `daily_activity` to merge with `sleep_day` data

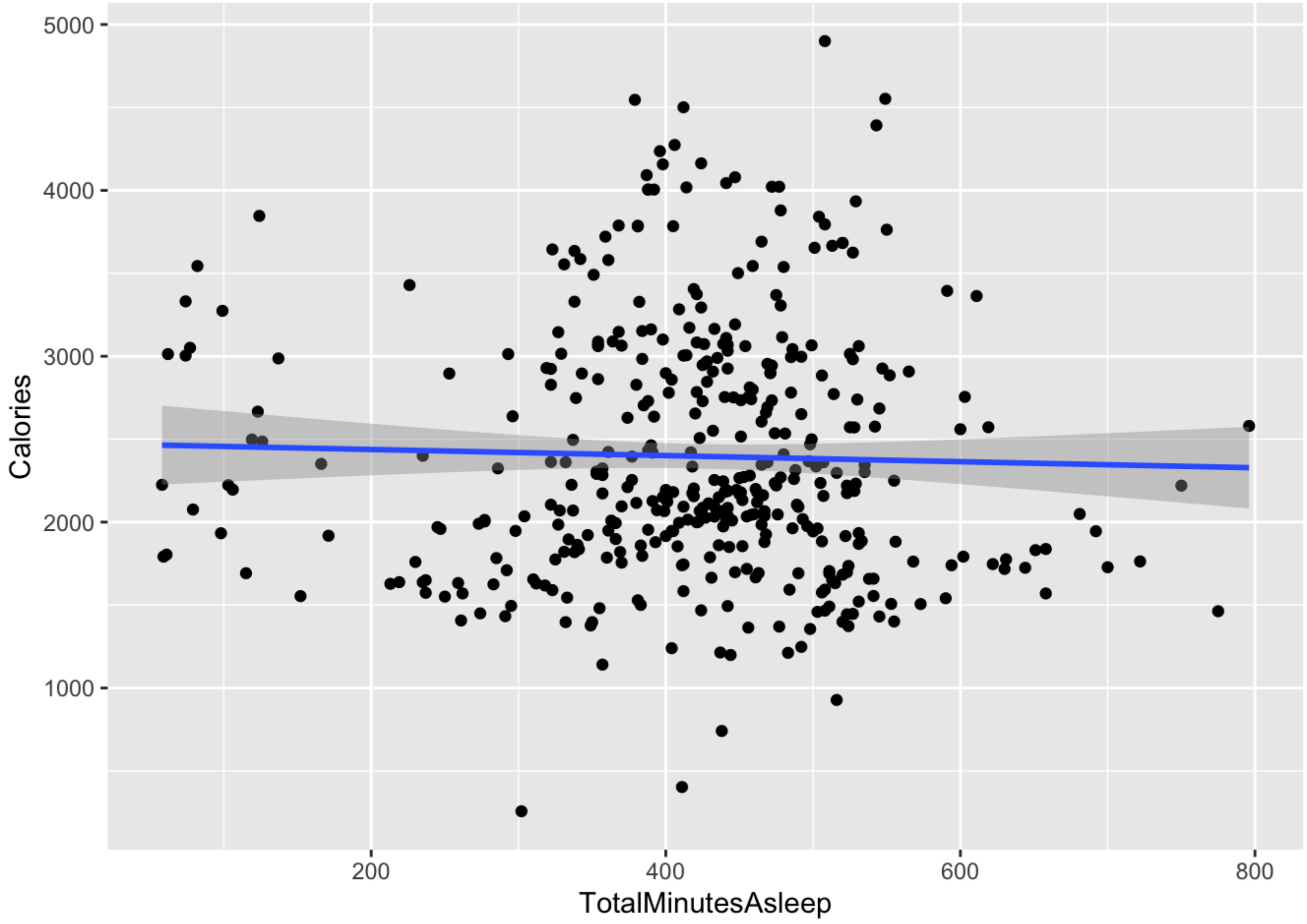
```
cal_daily = select(daily_activity, Id, ActivityDate, Calories, day_of_week)
```

Merging `cal_daily` which is sub set of `daily_activity` and `sleep_day`

```
sleep_data = merge(sleep_day, cal_daily, by=c('Id', 'ActivityDate', 'day_of_week'))
```

Visualizing sleep vs Calories data

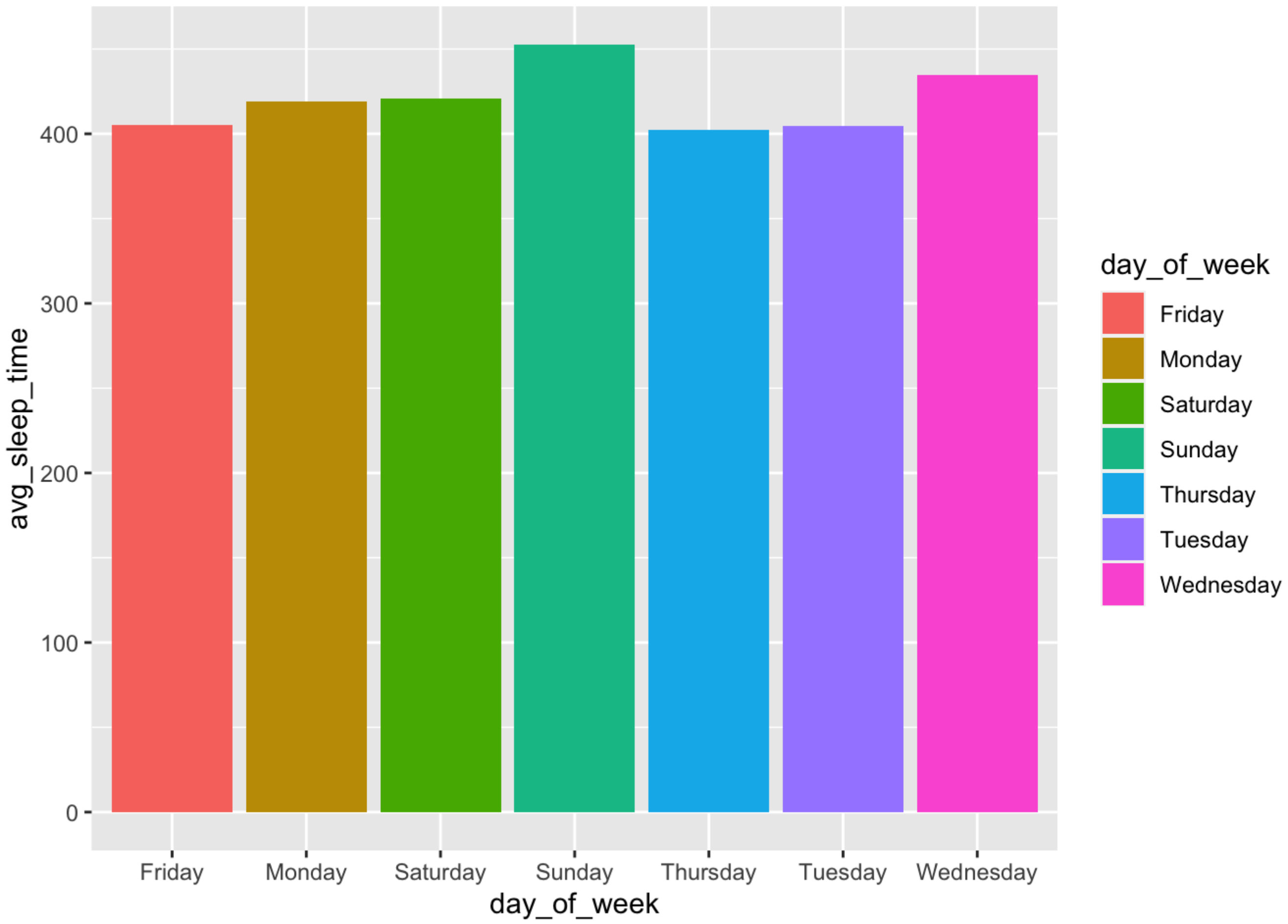
```
ggplot(sleep_data, aes(TotalMinutesAsleep, Calories)) + geom_point() +
  geom_smooth(method=lm)
```



It seems that their is no significant relation between sleep time and calories burned.

Average Sleep time in week

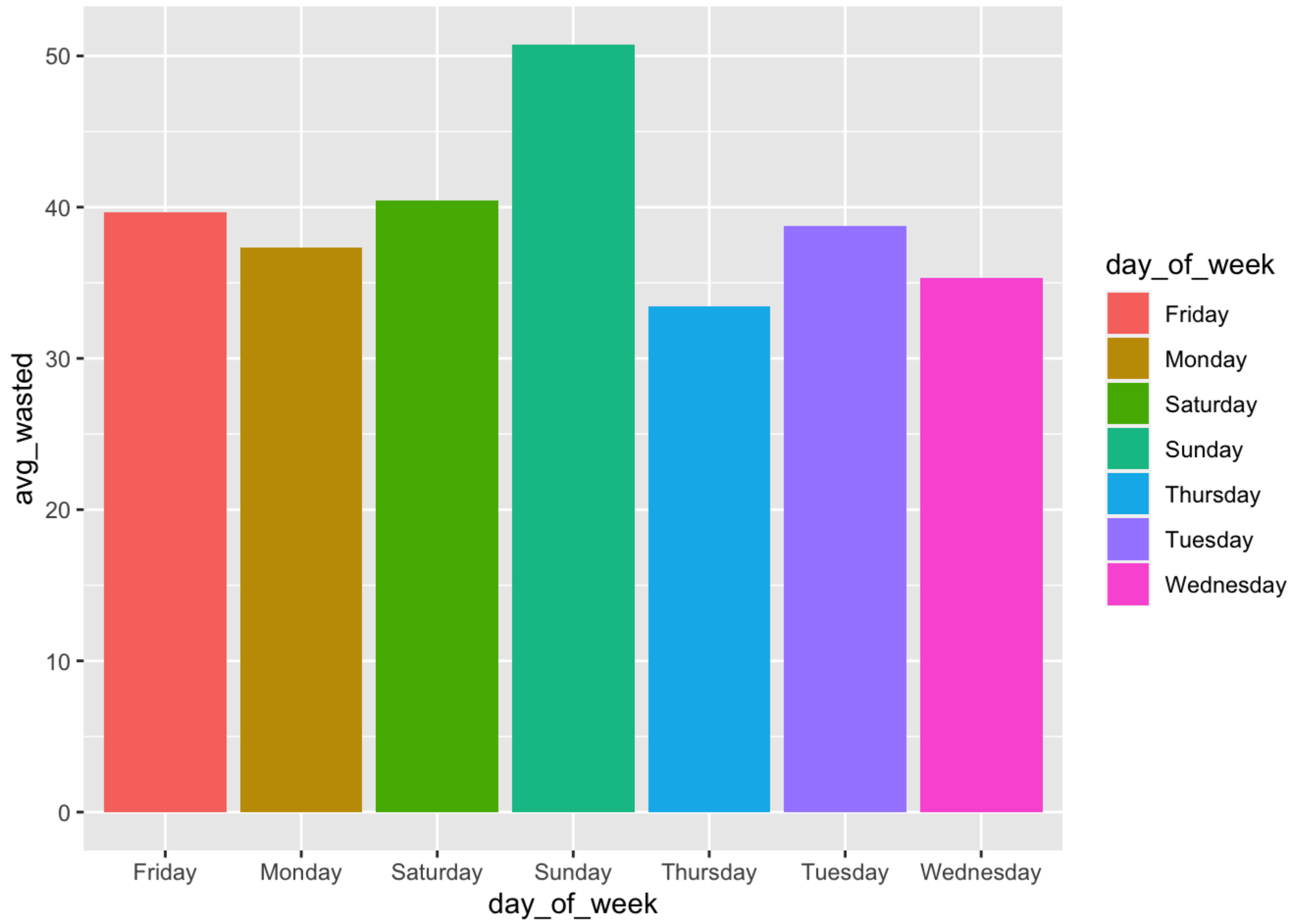
```
avg_sleep = sleep_data %>%
  group_by(day_of_week) %>%
  summarise(avg_sleep_time = mean(TotalMinutesAsleep))

ggplot(avg_sleep) + geom_col(mapping=aes(x=day_of_week, y=avg_sleep_time, fill=day_of
_week))
```



From the column chart we can see that Average sleep throughout the week is almost same, except Sunday , that’s probably because Sunday is weekend.

Average time wasted in bed after sleep

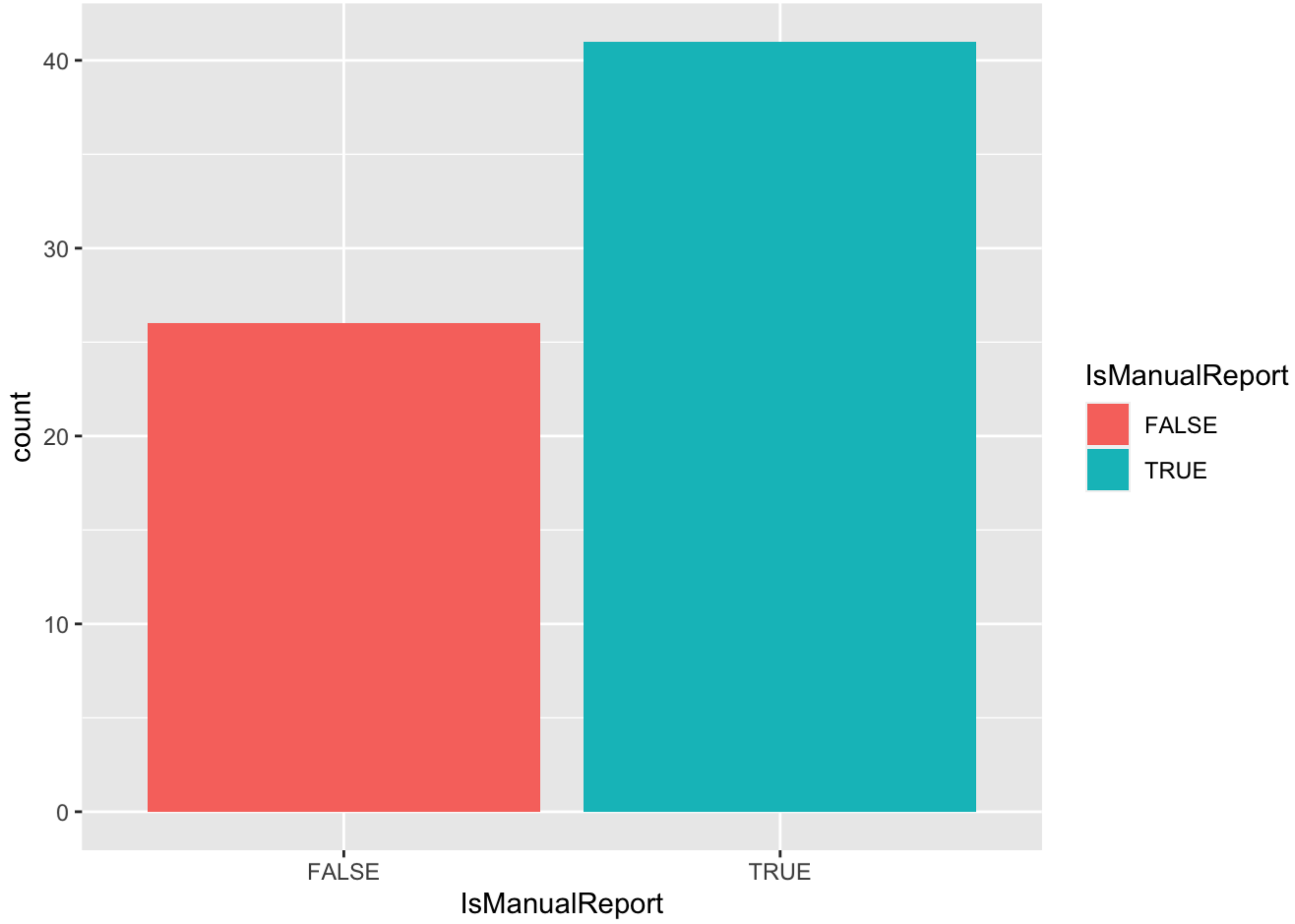
```
avg_wasted = sleep_data %>%
  mutate(time_wasted = TotalTimeInBed - TotalMinutesAsleep) %>%
  group_by(day_of_week) %>%
  summarise(avg_wasted = mean(time_wasted))

ggplot(avg_wasted) + geom_col(mapping=aes(x=day_of_week, y=avg_wasted, fill=day_of_we
ek))
```



From above visualization we can see that most of the users waste more then 50 minutes in bed when it is Sunday. During other weekdays wasted times are between 30-40 minutes.

Weight Log

```
ggplot(weight_log) + geom_bar(mapping=aes(x=IsManualReport, fill=IsManualReport))
```



We don’t have enough data to analyse weight log data deeply, and fat column also contains many NA values . But from our above bar chart we can see that maximum weight log inputs were done manually , which is not a good thing for a business and customer satisfaction.

Notable Points

- **Based on analyzing how Fitbit consumers use and respond to features, recommendations can be made to promote further growth for Bellabeat.**
- **Rather than simply providing data on user’s health, the app should further encourage users to meet fitness goals and become a social media platform**
- **Center of Disease Control recommends working out with a friend in order to feel motivated and be more adventurous in trying new workouts.**

Analysis Takeaways & Recommendation

- **Enable social networking system an application, so users can post their favorite workouts, wellness tips, healthy meals, and workout tips in the app.**
- **Enable users to add friends and view each other’s activity to create a competitive environment.**
- **Create weekly fitness and wellness challenges to encourage users , so that they workout a decent time in a week.**

- **Have health and fitness companies pay for advertising in the application.**
- **Recommend users to get 10,000 steps a day and enable alert notifications to encourage users to meet goal.**
- **Recommend users get at least 75 minutes of vigorous activity in a week and enable alert notifications to encourage users to meet recommended vigorous time .**
- **Encourage users to enter Age, height and Weight to track BMI .**
- **Alert users if their rest heart rate varies from normal.**
- **Set notification if a users spends more then an hour awake in bed .**
- **Alert users if they are spending more time then normal in sedentary state. Encourage them to do some activity rather then getting onto sedentary state.**
- **For weight log , the input is system is manual . It will be great if weight data can be Tracked and update automatically under Wifi connection.**

Recommendation for Bellabeat Membership

- **Offer 30-day free trial subscription.**
- **Offer reduced subscription fee when a member refers a friend.**
- **Offer discounts for Bellabeat smart device products with membership.**
- **Partner with health & fitness companies and offer discounts for members. Recommendations for Bellabeat products**
- **Offer a bundle deal for the Spring and Leaf together.**

Conclusion

Took lot of effort to make this whole report , and was kinda crazy idea to embedded html inside a R Mark Down `Report` , did it to customize however i want ;)

Thank You