BellaBeat

Exploratory Data Analysis for a Wellness Company

Kshitija Hire

Monday, August 30 2021

- DATA LOADING
    - Viewing the datasets
    - Checking the variables
- DATA CLEANING AND MANIPULATION
    - Adjusting the formats
    - Removing unwanted columns
- DATA EXPLORATION
    - Calories burned by Total steps taken
        - Dividing the distance data into three categories for easier analysis
        - Dividing the Total Steps data into three categories for easier analysis
    - Calories burned by Distance
    - Average Calories burned during the week
        - Let’s try and use the hourlyCalorie dataset
        - View the Dataset
        - Left join the two datasets
    - Average Calories burned per hour

Bellabeat is a high-tech manufacturer of health-focused products for women. Its goal is to become a large player in the global smart device market.

The data on which we are going to do our analysis has been collected from a BellaBeat smart device product. This product records physical activity, heart rate, and sleeping patterns.

We will divide our analysis in three categories: DATA LOADING, DATA CLEANING AND MANIPULATION AND DATA EXPLORATION

```
#install.packages("tidyverse")
```

DATA LOADING

```
library(tidyverse)
library(lubridate)
library(plotly)
```

Viewing the datasets

```
daily_activity <- read_csv("data/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.
```

```
hourly_calories <- read_csv("data/hourlyCalories_merged.csv")
```

```
## Rows: 22099 Columns: 3
```

```
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityHour
## dbl (2): Id, Calories
```

```
##
## 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.
```

```
hourly_steps <- read_csv("data/hourlySteps_merged.csv")
```

```
## Rows: 22099 Columns: 3
```

```
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityHour
## dbl (2): Id, StepTotal
```

```
##
## 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.
```

```
head(daily_activity, n=10)
```

```
## # A tibble: 10 × 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance LoggedActivitie…
##        <dbl> <chr>              <dbl>         <dbl>           <dbl>             <dbl>
##  1 1503960366 4/12/2016          13162          8.5             8.5                 0
##  2 1503960366 4/13/2016          10735          6.97            6.97                0
##  3 1503960366 4/14/2016          10460          6.74            6.74                0
##  4 1503960366 4/15/2016           9762          6.28            6.28                0
##  5 1503960366 4/16/2016          12669          8.16            8.16                0
##  6 1503960366 4/17/2016           9705          6.48            6.48                0
##  7 1503960366 4/18/2016          13019          8.59            8.59                0
##  8 1503960366 4/19/2016          15506          9.88            9.88                0
##  9 1503960366 4/20/2016          10544          6.68            6.68                0
## 10 1503960366 4/21/2016           9819          6.34            6.34                0
## # … with 9 more variables: VeryActiveDistance <dbl>,
## #   ModeratelyActiveDistance <dbl>, LightActiveDistance <dbl>,
## #   SedentaryActiveDistance <dbl>, VeryActiveMinutes <dbl>,
## #   FairlyActiveMinutes <dbl>, LightlyActiveMinutes <dbl>,
## #   SedentaryMinutes <dbl>, Calories <dbl>
```

Checking the variables

```
str(daily_activity)
```

```
## spec_tbl_df [940 × 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" "4/15/201
6" ...
##  $ 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>
```

DATA CLEANING AND MANIPULATION

Adjusting the formats

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

daily_activity
```

```
## # A tibble: 940 × 15
##             Id Date       TotalSteps TotalDistance TrackerDistance LoggedActivitie…
##          <dbl> <date>          <dbl>         <dbl>           <dbl>             <dbl>
##  1 1503960366 2016-04-12      13162          8.5             8.5                 0
##  2 1503960366 2016-04-13      10735          6.97            6.97                0
##  3 1503960366 2016-04-14      10460          6.74            6.74                0
##  4 1503960366 2016-04-15       9762          6.28            6.28                0
##  5 1503960366 2016-04-16      12669          8.16            8.16                0
##  6 1503960366 2016-04-17       9705          6.48            6.48                0
##  7 1503960366 2016-04-18      13019          8.59            8.59                0
##  8 1503960366 2016-04-19      15506          9.88            9.88                0
##  9 1503960366 2016-04-20      10544          6.68            6.68                0
## 10 1503960366 2016-04-21       9819          6.34            6.34                0
## # … with 930 more rows, and 9 more variables: VeryActiveDistance <dbl>,
## #   ModeratelyActiveDistance <dbl>, LightActiveDistance <dbl>,
## #   SedentaryActiveDistance <dbl>, VeryActiveMinutes <dbl>,
## #   FairlyActiveMinutes <dbl>, LightlyActiveMinutes <dbl>,
## #   SedentaryMinutes <dbl>, Calories <dbl>
```

Removing unwanted columns

```
# Dropping unwanted columns
daily_activity <-  daily_activity %>%
  select(-c(TrackerDistance, SedentaryActiveDistance, LoggedActivitiesDistance, VeryActive
Distance:SedentaryMinutes ))
daily_activity
```

```
## # A tibble: 940 × 5
##             Id Date       TotalSteps TotalDistance Calories
##          <dbl> <date>          <dbl>         <dbl>    <dbl>
##  1 1503960366 2016-04-12      13162          8.5      1985
##  2 1503960366 2016-04-13      10735          6.97     1797
##  3 1503960366 2016-04-14      10460          6.74     1776
##  4 1503960366 2016-04-15       9762          6.28     1745
##  5 1503960366 2016-04-16      12669          8.16     1863
##  6 1503960366 2016-04-17       9705          6.48     1728
##  7 1503960366 2016-04-18      13019          8.59     1921
##  8 1503960366 2016-04-19      15506          9.88     2035
##  9 1503960366 2016-04-20      10544          6.68     1786
## 10 1503960366 2016-04-21       9819          6.34     1775
## # … with 930 more rows
```

```
#Calculate unique number of participants
daily_activity %>% count(Id)
```

```
## # A tibble: 33 × 2
##            Id     n
##         <dbl> <int>
##  1 1503960366    31
##  2 1624580081    31
##  3 1644430081    30
##  4 1844505072    31
##  5 1927972279    31
##  6 2022484408    31
##  7 2026352035    31
##  8 2320127002    31
##  9 2347167796    18
## 10 2873212765    31
## # … with 23 more rows
```

We see that we have a data of 33 women who recorded their daily activity using Bella Beat smart device product for 31 days.

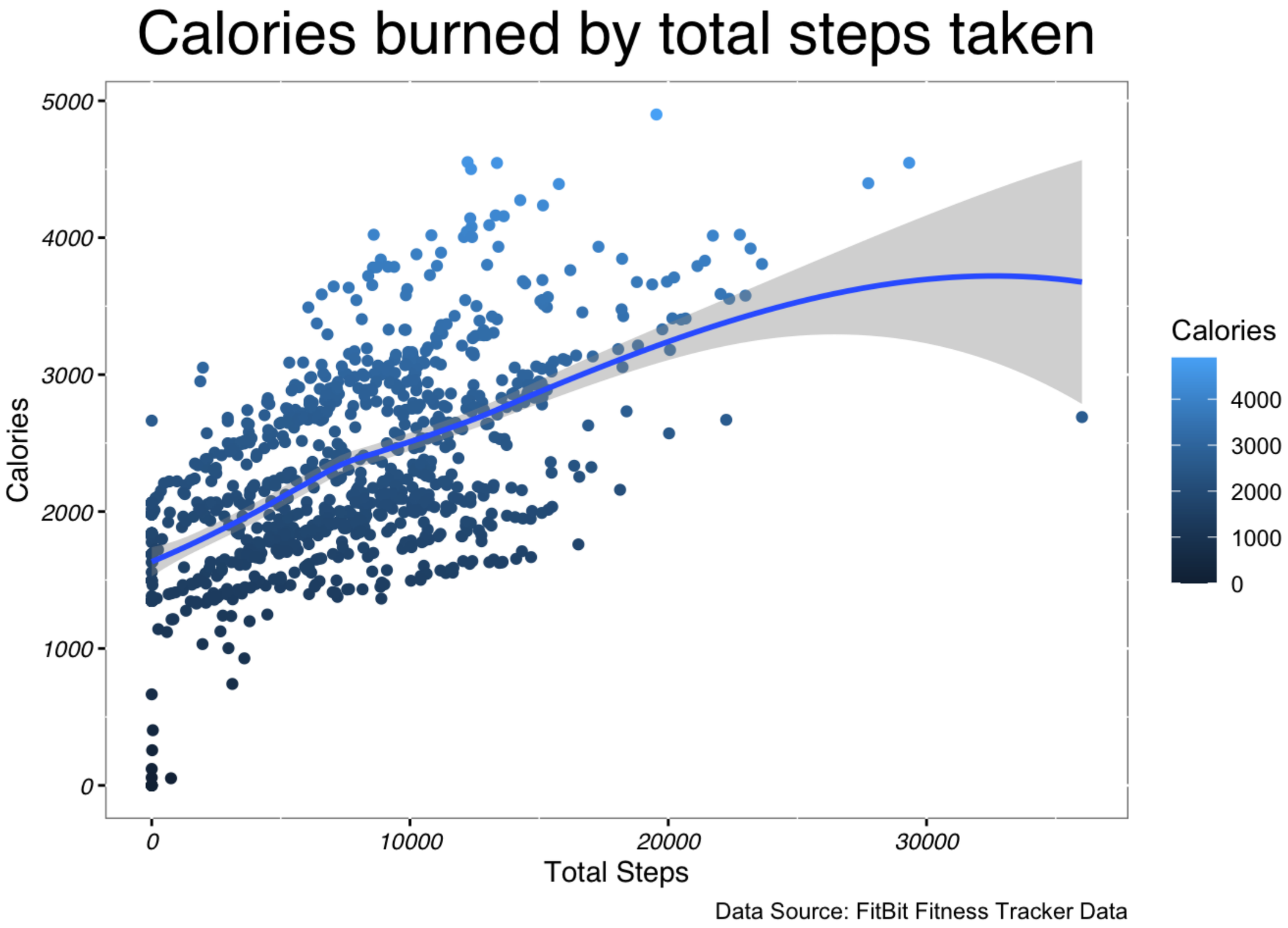
DATA EXPLORATION

Calories burned by Total steps taken

```
custom_theme <- function() {
  theme(
     panel.background = element_rect(fill = "white", color = 'grey50'),
     axis.text = element_text(colour = "black",
                              face = "italic",
                              family = "Helvetica"),
    axis.title = element_text(colour = "black",
                              family = "Helvetica"),
    axis.ticks = element_line(colour = "black"),
    plot.title = element_text(size=23,
                              hjust = 0.5,
                              family = "Helvetica")
  )
}
```

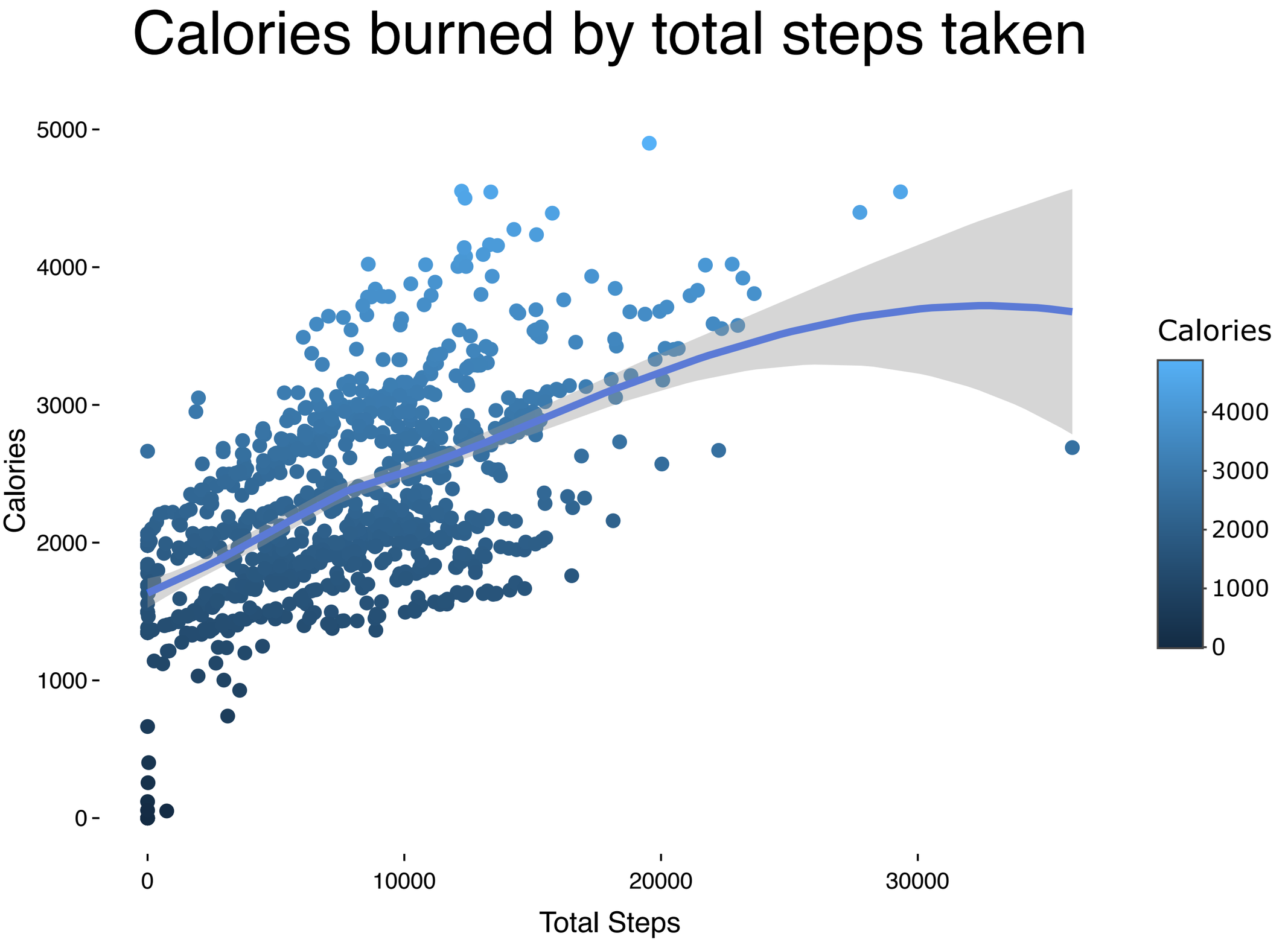
```
daily_activity %>%
  ggplot(aes(x = TotalSteps,
             y = Calories)) +
  geom_point(aes(colour = Calories))+
  geom_smooth()+
  custom_theme()+
  labs(title = 'Calories burned by total steps taken',
       y = 'Calories',
       x = 'Total Steps',
       caption = 'Data Source: FitBit Fitness Tracker Data')
```

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




```
ggplotly()
```

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



There is an evident positive relation between the total number of steps taken by a participant and the calories burned by them. But, it is not the only reason for their burning calories as the plot is quite scattered.

For example, Lets take 10,000 steps. The calories burned for one participant are 1500 and for other, it is 4000. Which is a huge difference.

Let’s explore in detail what other factors come into play into burning more calories for the same number of steps taken.

Dividing the distance data into three categories for easier analysis

```
daily_activity <- daily_activity %>%
  drop_na() %>%
  mutate(Dist_Category = ifelse(TotalDistance < 4, 'Less than 4 miles',
                         ifelse(TotalDistance >= 4 & TotalDistance <= 7, 'Between 4 and 7
 miles',
                         'More than 7 miles'))) %>%
mutate(Dist_Category = factor(Dist_Category, levels = c("Less than 4 miles", "Between 4 an
d 7 miles", "More than 7 miles")))

daily_activity
```

```
## # A tibble: 940 × 6
##            Id Date       TotalSteps TotalDistance Calories Dist_Category
##         <dbl> <date>          <dbl>         <dbl>    <dbl> <fct>
##  1 1503960366 2016-04-12      13162          8.5      1985 More than 7 miles
##  2 1503960366 2016-04-13      10735          6.97     1797 Between 4 and 7 miles
##  3 1503960366 2016-04-14      10460          6.74     1776 Between 4 and 7 miles
##  4 1503960366 2016-04-15       9762          6.28     1745 Between 4 and 7 miles
##  5 1503960366 2016-04-16      12669          8.16     1863 More than 7 miles
##  6 1503960366 2016-04-17       9705          6.48     1728 Between 4 and 7 miles
##  7 1503960366 2016-04-18      13019          8.59     1921 More than 7 miles
##  8 1503960366 2016-04-19      15506          9.88     2035 More than 7 miles
##  9 1503960366 2016-04-20      10544          6.68     1786 Between 4 and 7 miles
## 10 1503960366 2016-04-21       9819          6.34     1775 Between 4 and 7 miles
## # … with 930 more rows
```

Dividing the Total Steps data into three categories for easier analysis

```
daily_activity <- daily_activity %>%
  drop_na() %>%
  mutate(TotalSteps_Category = ifelse(TotalSteps < 6000, 'Less than 6k steps',
                          ifelse(TotalSteps >= 6000 & TotalSteps <= 10000, 'Between 6k and
 10k steps',
                          'More than 10k'))) %>%
mutate(TotalSteps_Category = factor(TotalSteps_Category, levels = c("Less than 6k steps",
"Between 6k and 10k steps", "More than 10k")))

daily_activity
```

```
## # A tibble: 940 × 7
##            Id Date       TotalSteps TotalDistance Calories Dist_Category
##         <dbl> <date>          <dbl>         <dbl>    <dbl> <fct>
##  1 1503960366 2016-04-12      13162          8.5      1985 More than 7 miles
##  2 1503960366 2016-04-13      10735          6.97     1797 Between 4 and 7 miles
##  3 1503960366 2016-04-14      10460          6.74     1776 Between 4 and 7 miles
##  4 1503960366 2016-04-15       9762          6.28     1745 Between 4 and 7 miles
##  5 1503960366 2016-04-16      12669          8.16     1863 More than 7 miles
##  6 1503960366 2016-04-17       9705          6.48     1728 Between 4 and 7 miles
##  7 1503960366 2016-04-18      13019          8.59     1921 More than 7 miles
##  8 1503960366 2016-04-19      15506          9.88     2035 More than 7 miles
##  9 1503960366 2016-04-20      10544          6.68     1786 Between 4 and 7 miles
## 10 1503960366 2016-04-21       9819          6.34     1775 Between 4 and 7 miles
## # … with 930 more rows, and 1 more variable: TotalSteps_Category <fct>
```

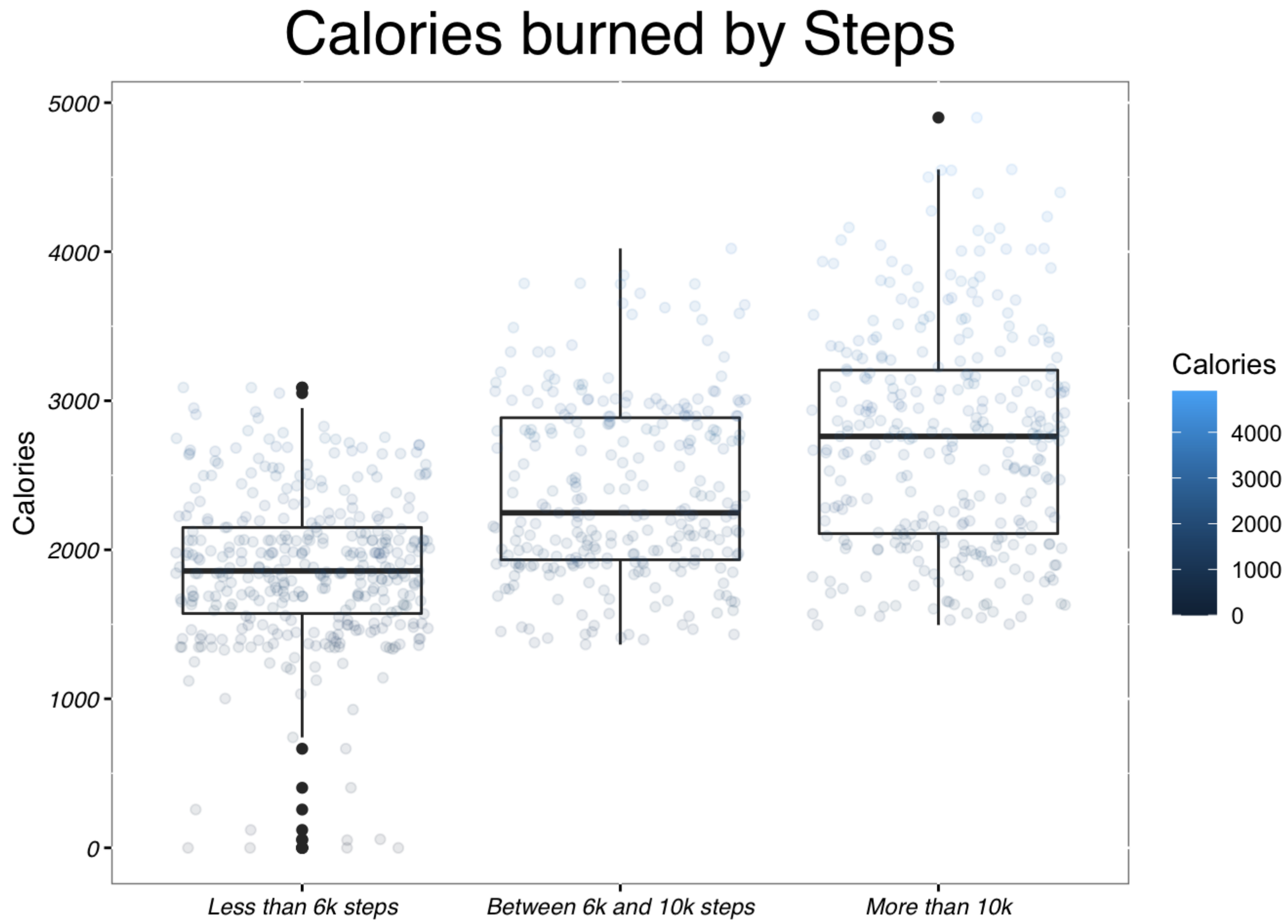
```
summary(daily_activity)
```

```
##        Id                  Date              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
##     Calories                    Dist_Category              TotalSteps_Category
##  Min.   :   0   Less than 4 miles    :354   Less than 6k steps      :366
##  1st Qu.:1828   Between 4 and 7 miles:282   Between 6k and 10k steps:271
##  Median :2134   More than 7 miles    :304   More than 10k           :303
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900
```

We could observe : The average women seems to walk more number of steps but eventually cover less distance.

Before moving on to additional factors involved in burning more calories with less or the same number of steps, let's take another look at the link between Calories Burned and Steps Taken.

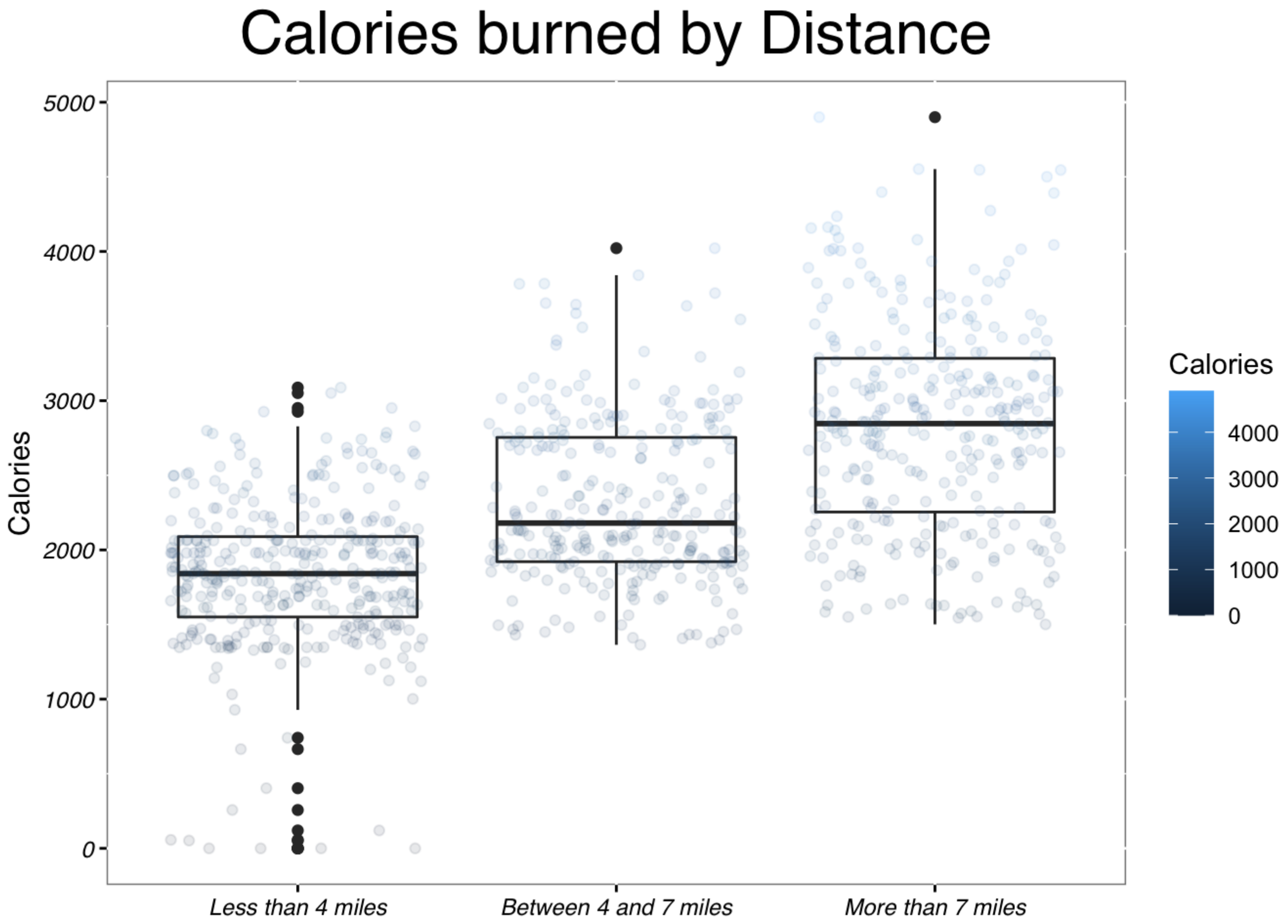
```
daily_activity %>%
ggplot(aes(TotalSteps_Category,Calories)) +
  geom_boxplot() +
  geom_jitter(alpha = 0.1, aes(colour = Calories))+
  custom_theme()+

  labs(title="Calories burned by Steps",x=NULL)
```



There is an even distribution of women walking from less than 6k steps to more than 10k.

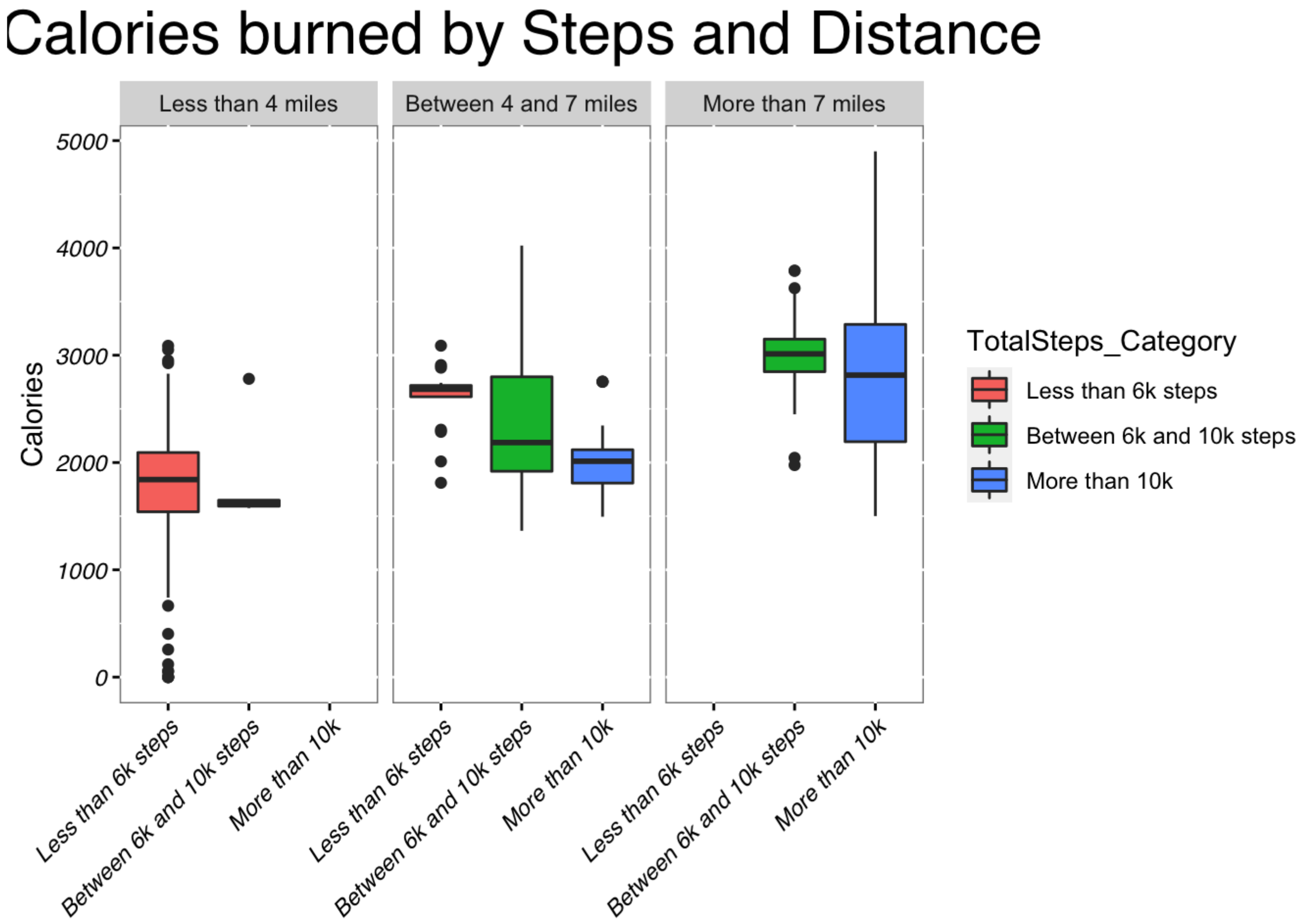
Calories burned by Distance

```
daily_activity %>%
ggplot(aes(Dist_Category,Calories)) +
  geom_boxplot() +
  geom_jitter(alpha = 0.1,aes(colour = Calories))+
  custom_theme()+

    labs(title="Calories burned by Distance",x=NULL)
```



We can see that when women walk longer distances, they burn more calories.

```
daily_activity %>%
ggplot(aes(TotalSteps_Category,Calories,fill=TotalSteps_Category)) +
  geom_boxplot() +
    facet_wrap(~Dist_Category)+
  custom_theme()+

    labs(title="Calories burned by Steps and Distance",x=NULL) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Different distance, Different number of steps:

“More than 10k steps” in “between 4 and 7 miles” and “less than 6k steps” in “less than 4 miles” both burn the same amount of calories.

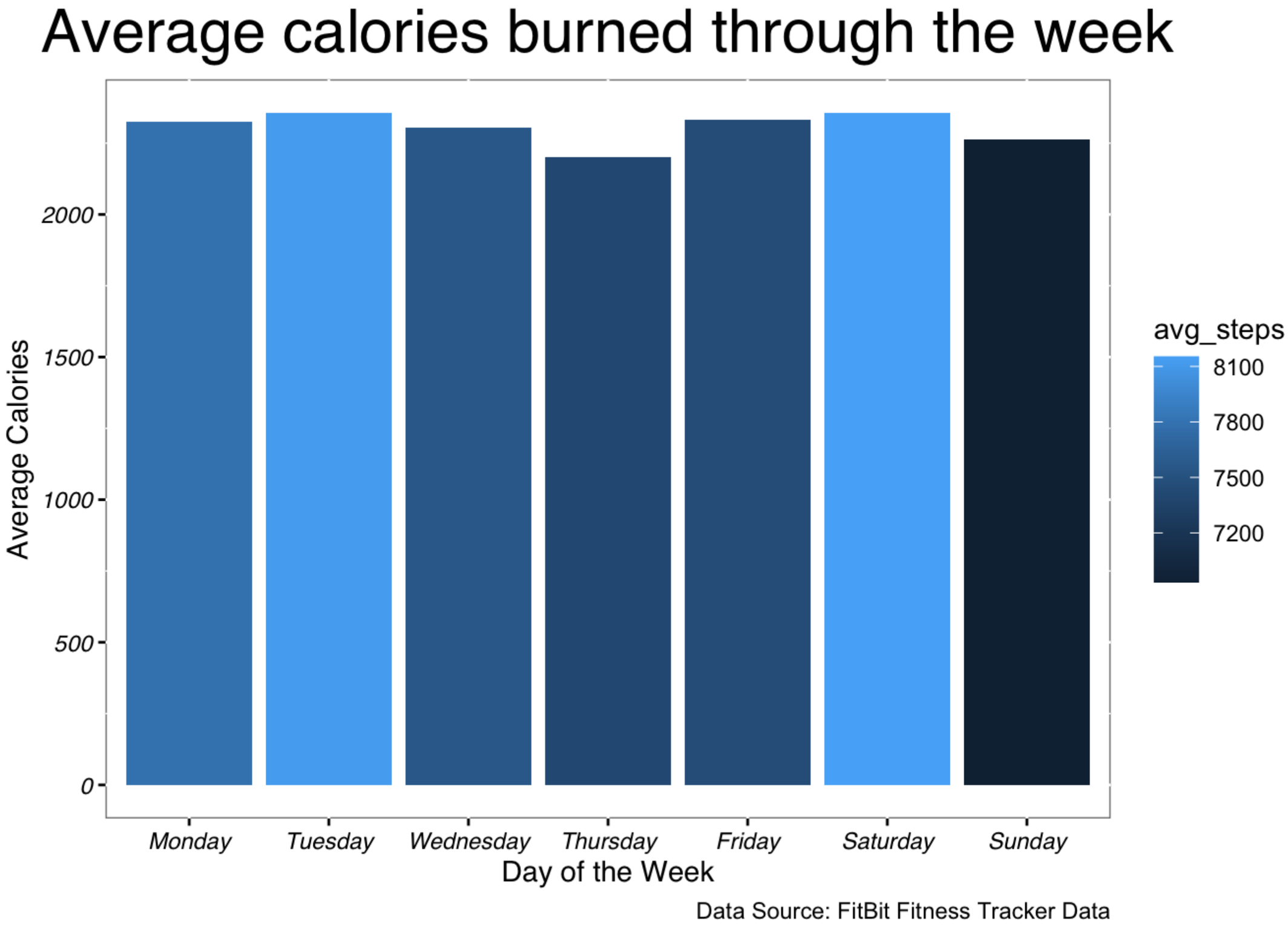
Same distance, Different number of steps:

In “More than 7 miles”, more calories are burned with less number of steps (between 6-10k) than more than 10k.

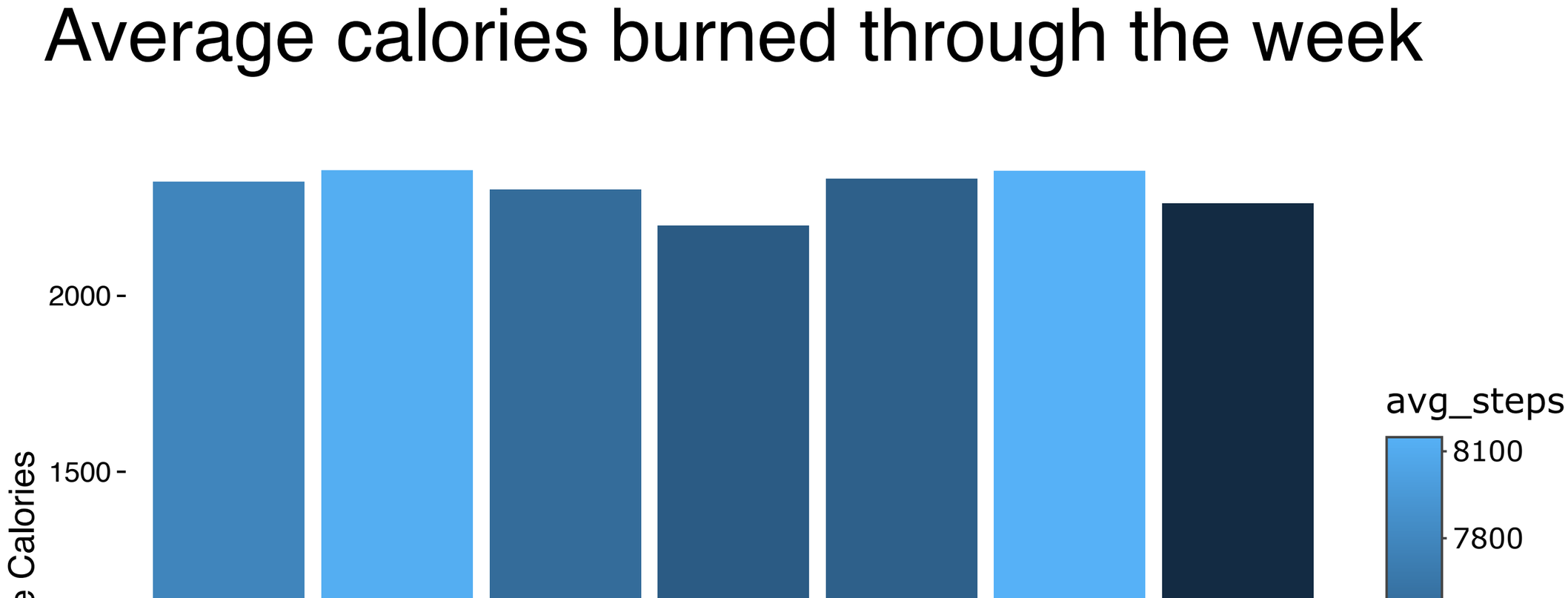
This reinforces the idea that speed is one of the most important factor to burn calories.

Average Calories burned during the week

```
daily_activity %>%
  mutate(weekdays = weekdays(Date)) %>%
  mutate(weekdays = factor(weekdays, levels = c('Monday','Tuesday', 'Wednesday', 'Thursda
y', 'Friday', 'Saturday', 'Sunday' ))) %>%
  select(weekdays, TotalSteps, Calories) %>%
group_by(weekdays) %>%
summarise(avg_cal = mean(Calories, na.rm = TRUE),
          avg_steps = mean(TotalSteps, na.rm = TRUE)) %>%
ggplot(aes(x = weekdays, y = avg_cal))+
geom_col(aes(fill = avg_steps))+

custom_theme()+
 labs(title = 'Average calories burned through the week',
     y = 'Average Calories',
     x = 'Day of the Week',
     caption = 'Data Source: FitBit Fitness Tracker Data',
     legend = 'Average Steps')
```




```
ggplotly()
```



The participating women seems to be fairly active throughout the week. The maximum number of average Calories burned are on Sunday followed by Thursday which is equally surprising.

To add in some more information, I have added the average steps taken each day of the week and the result is predictable.

Since, there is not much of a difference in average calories throughout the week, we should narrow it down to each day by hour to try and see some pattern.

Let's try and use the hourlyCalorie dataset

```
#Formatting the Date column
hourly_calories <- hourly_calories %>%
  rename(DateTime = ActivityHour) %>%
  mutate(DateTime = as_datetime(DateTime, format="%m/%d/%Y %I:%M:%S %p"))

#Adding seperate Date column
  hourly_calories$Date <- as.Date(hourly_calories$DateTime)

#Adding seperate Time Column
  hourly_calories$Time <- format(hourly_calories$DateTime,format = "%H:%M:%S")
```

View the Dataset

```
str(hourly_calories)
```

```
## spec_tbl_df [22,099 × 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Id      : num [1:22099] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ DateTime: POSIXct[1:22099], format: "2016-04-12 00:00:00" "2016-04-12 01:00:00" ...
##  $ Calories: num [1:22099] 81 61 59 47 48 48 48 47 68 141 ...
##  $ Date    : Date[1:22099], format: "2016-04-12" "2016-04-12" ...
##  $ Time    : chr [1:22099] "00:00:00" "01:00:00" "02:00:00" "03:00:00" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   ActivityHour = col_character(),
##   ..   Calories = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Left join the two datasets

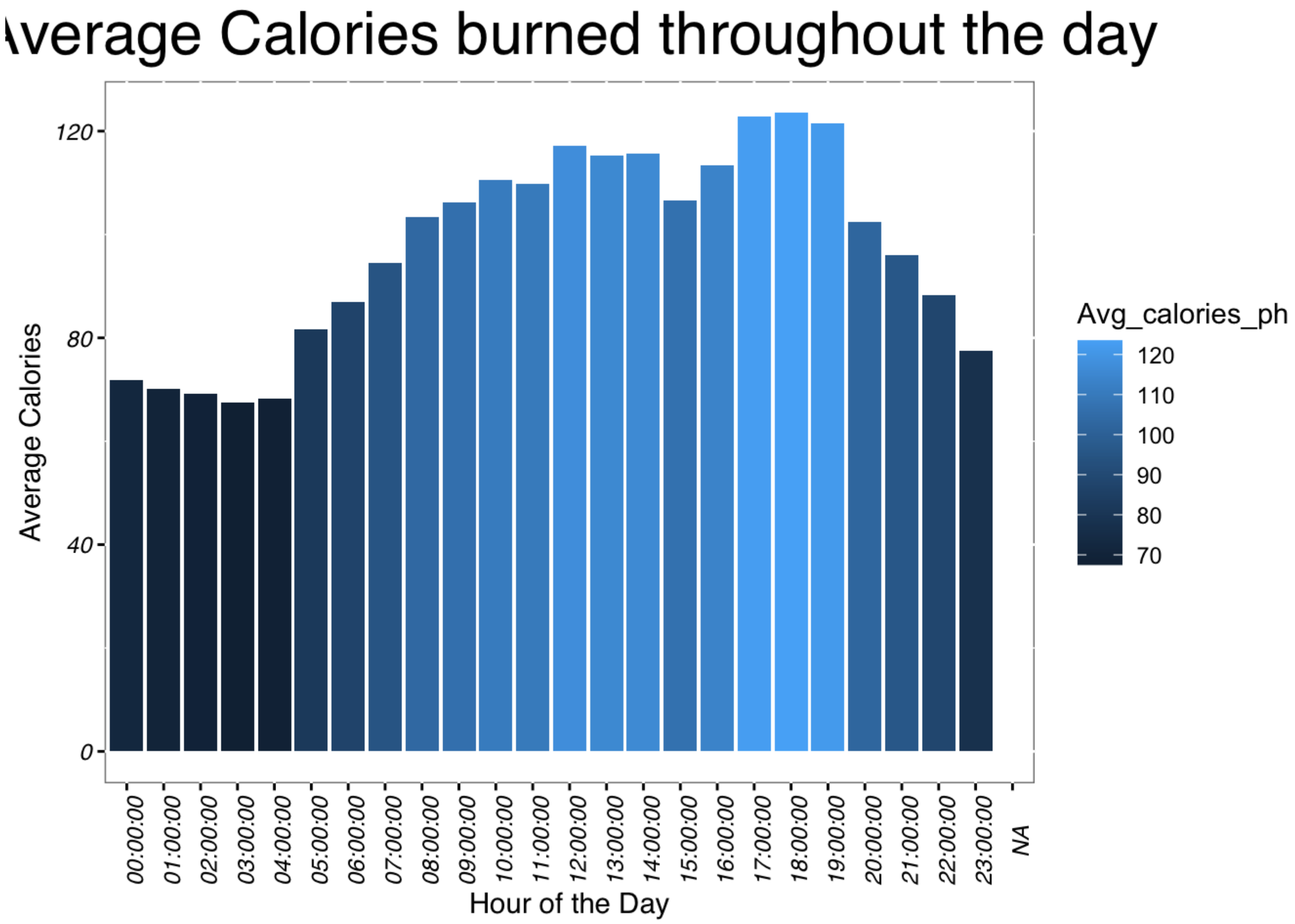
```
merge1 <- left_join(daily_activity, hourly_calories, by = c('Id','Date'))
merge1
```

```
## # A tibble: 22,105 × 10
##            Id Date       TotalSteps TotalDistance Calories.x Dist_Category
##         <dbl> <date>          <dbl>         <dbl>      <dbl> <fct>
##  1 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  2 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  3 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  4 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  5 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  6 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  7 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  8 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
##  9 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
## 10 1503960366 2016-04-12      13162           8.5       1985 More than 7 miles
## # … with 22,095 more rows, and 4 more variables: TotalSteps_Category <fct>,
## #   DateTime <dttm>, Calories.y <dbl>, Time <chr>
```

Average Calories burned per hour

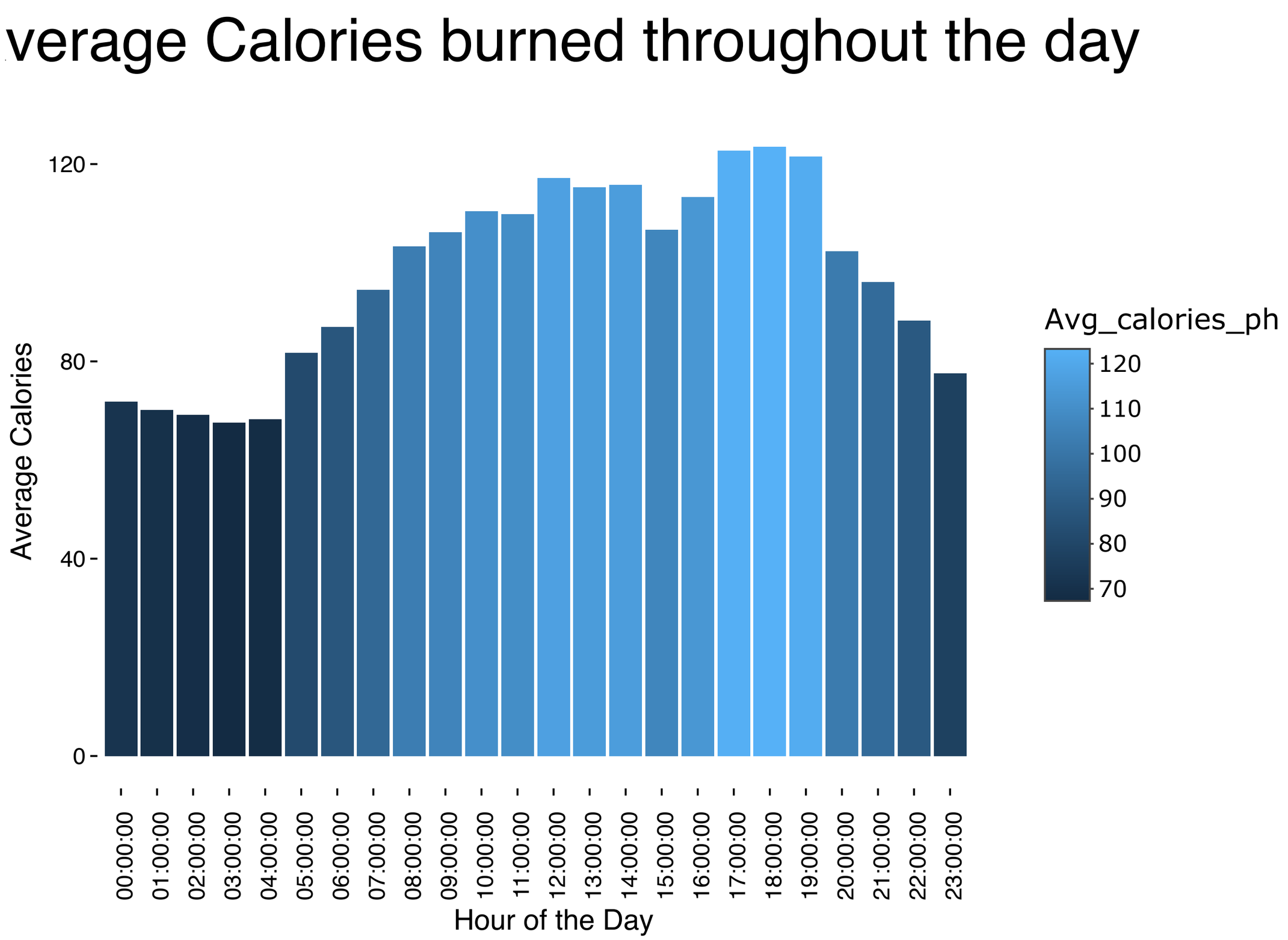
```
merge1 %>%
  select(Time, Calories.y, ) %>%
  group_by(Time) %>%
  summarise(Avg_calories_ph = mean(Calories.y, na.rm = TRUE)) %>%
  ggplot(aes(x = Time,
             y = Avg_calories_ph)) +
  geom_col(aes(fill = Avg_calories_ph))+
  theme(axis.text.x = element_text(angle = 90)) +
  custom_theme()+
  labs(x = "Hour of the Day",
       y = "Average Calories",
       title ="Average Calories burned throughout the day")
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```




```
ggplotly()
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```



We can see that maximum average number of calories are burned in the evening (5 pm to 7 pm) and during the lunch hour(12pm to 2pm).