FitnessTracker

2024-01-21

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(here)
```

```
## here() starts at /cloud/project
```

```
library(skimr)
library(janitor)
```

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

```
library(lubridate)
library(ggplot2)
```

#read the files

```
daily_activity <- read_csv("dailyActivity.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.
```

```
#View(daily_activity)

sleepDay <- read_csv("sleepDay.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.
```

```
#View(sleepDay)

hourlyIntensities<-read_csv("hourlyIntensities.csv")
```

```
## Rows: 22099 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): ActivityHour
## dbl (3): Id, TotalIntensity, AverageIntensity
##
## 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.
```

```
#View(hourlyIntensities)
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

#Fixing formatting

```
#SleepDay
sleepDay$SleepDay = as.POSIXct(sleepDay$SleepDay, format = "%m/%d/%Y %H:%M:%S", tz = Sys.timezone())
sleepDay$date<-format(sleepDay$SleepDay,format="%m/%d/%Y")
sleepDay$time<-format(sleepDay$SleepDay,format="%H:%M:%S")
```

```
#hourlyintensities
hourlyIntensities$ActivityHour=as.POSIXct(hourlyIntensities$ActivityHour, format="%m/%d/%Y %I:%M:%S %p"
hourlyIntensities$time <- format(hourlyIntensities$ActivityHour, format = "%H:%M:%S")
hourlyIntensities$date <- format(hourlyIntensities$ActivityHour, format = "%m/%d/%y")
```

##Preview the data

```
#dailyActivity
head(daily_activity)
```

```
## # A tibble: 6 x 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
##        <dbl> <chr>             <dbl>         <dbl>           <dbl>
## 1 1503960366 4/12/2016         13162          8.5             8.5
## 2 1503960366 4/13/2016         10735          6.97            6.97
## 3 1503960366 4/14/2016         10460          6.74            6.74
## 4 1503960366 4/15/2016          9762          6.28            6.28
## 5 1503960366 4/16/2016         12669          8.16            8.16
## 6 1503960366 4/17/2016          9705          6.48            6.48
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

```r
str(daily_activity)
```

```
## spc_tbl_ [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>
```

```r
head(hourlyIntensities)
```

```
## # A tibble: 6 x 6
##           Id ActivityHour        TotalIntensity AverageIntensity time     date
##        <dbl> <dttm>                       <dbl>            <dbl> <chr>    <chr>
## 1 1503960366 2016-04-12 00:00:00             20            0.333 00:00:00 04/12~
## 2 1503960366 2016-04-12 01:00:00              8            0.133 01:00:00 04/12~
## 3 1503960366 2016-04-12 02:00:00              7            0.117 02:00:00 04/12~
## 4 1503960366 2016-04-12 03:00:00              0            0     03:00:00 04/12~
## 5 1503960366 2016-04-12 04:00:00              0            0     04:00:00 04/12~
## 6 1503960366 2016-04-12 05:00:00              0            0     05:00:00 04/12~
```

```r
#SleepDay
str(sleepDay)
```

```
## spc_tbl_ [413 x 7] (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          : POSIXct[1:413], format: "2016-04-12 12:00:00" "2016-04-13 12:00:00" ...
##  $ TotalSleepRecords : num [1:413] 1 2 1 2 1 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 ...
##  $ date              : chr [1:413] "04/12/2016" "04/13/2016" "04/15/2016" "04/16/2016" ...
##  $ time              : chr [1:413] "12:00:00" "12:00:00" "12:00:00" "12:00:00" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   SleepDay = col_character(),
##   ..   TotalSleepRecords = col_double(),
##   ..   TotalMinutesAsleep = col_double(),
##   ..   TotalTimeInBed = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

#finding distinct users in each data frame

```
n_distinct(daily_activity$Id)
```

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

```
n_distinct(sleepDay$Id)
```

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

#summary statistics

```
daily_activity %>%
  select(TotalDistance,
         VeryActiveDistance,
         SedentaryActiveDistance,
         LightActiveDistance,
         VeryActiveMinutes,
         SedentaryMinutes,
         LightlyActiveMinutes) %>%
  summary()
```

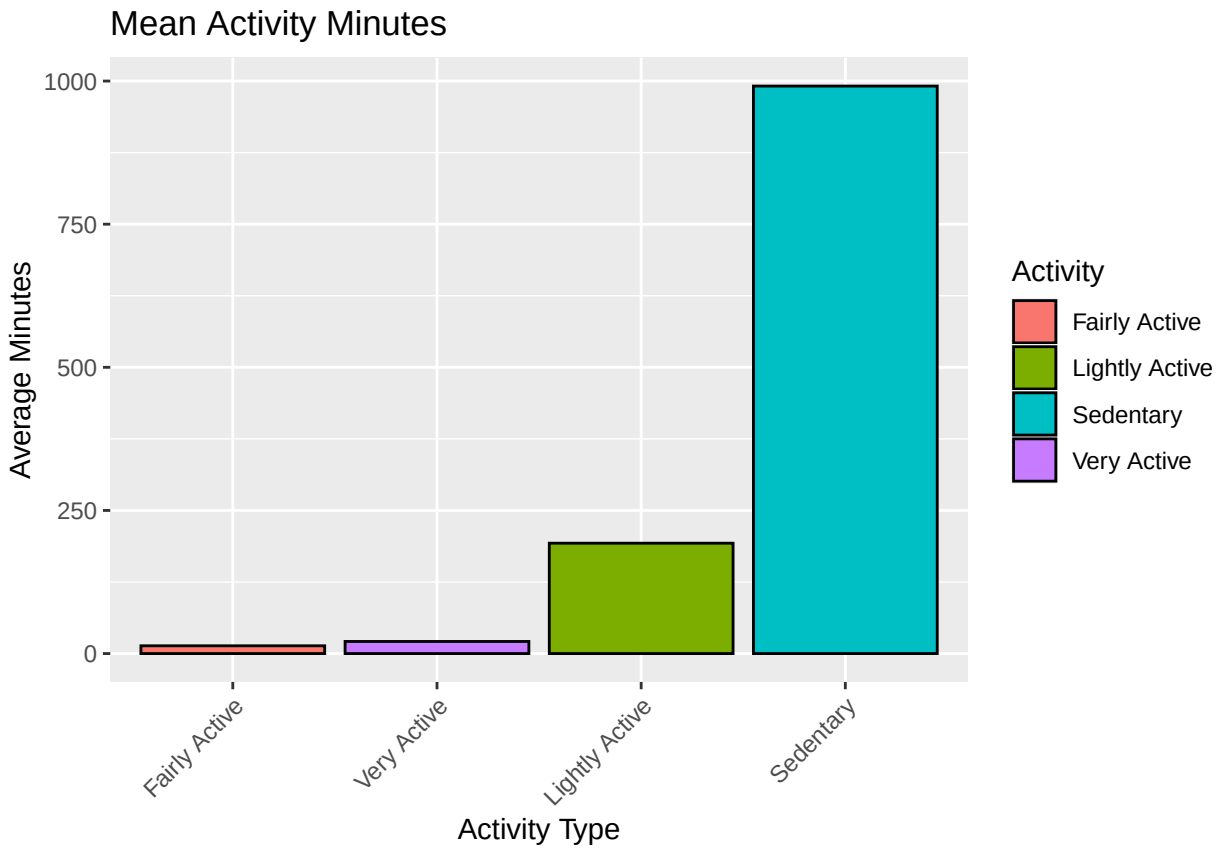
```
##  TotalDistance    VeryActiveDistance SedentaryActiveDistance
##  Min.   : 0.000   Min.   : 0.000     Min.   :0.000000
##  1st Qu.: 2.620   1st Qu.: 0.000     1st Qu.:0.000000
##  Median : 5.245   Median : 0.210     Median :0.000000
##  Mean   : 5.490   Mean   : 1.503     Mean   :0.001606
##  3rd Qu.: 7.713   3rd Qu.: 2.053     3rd Qu.:0.000000
##  Max.   :28.030   Max.   :21.920     Max.   :0.110000
##  LightActiveDistance VeryActiveMinutes SedentaryMinutes LightlyActiveMinutes
##  Min.   : 0.000      Min.   :  0.00    Min.   :   0.0   Min.   :  0.0
##  1st Qu.: 1.945      1st Qu.:  0.00    1st Qu.: 729.8   1st Qu.:127.0
##  Median : 3.365      Median :  4.00    Median :1057.5   Median :199.0
##  Mean   : 3.341      Mean   : 21.16    Mean   : 991.2   Mean   :192.8
##  3rd Qu.: 4.782      3rd Qu.: 32.00    3rd Qu.:1229.5   3rd Qu.:264.0
##  Max.   :10.710      Max.   :210.00    Max.   :1440.0   Max.   :518.0
```

```
sleepDay %>%
  select(TotalMinutesAsleep,
         TotalTimeInBed) %>%
  summary()
```

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

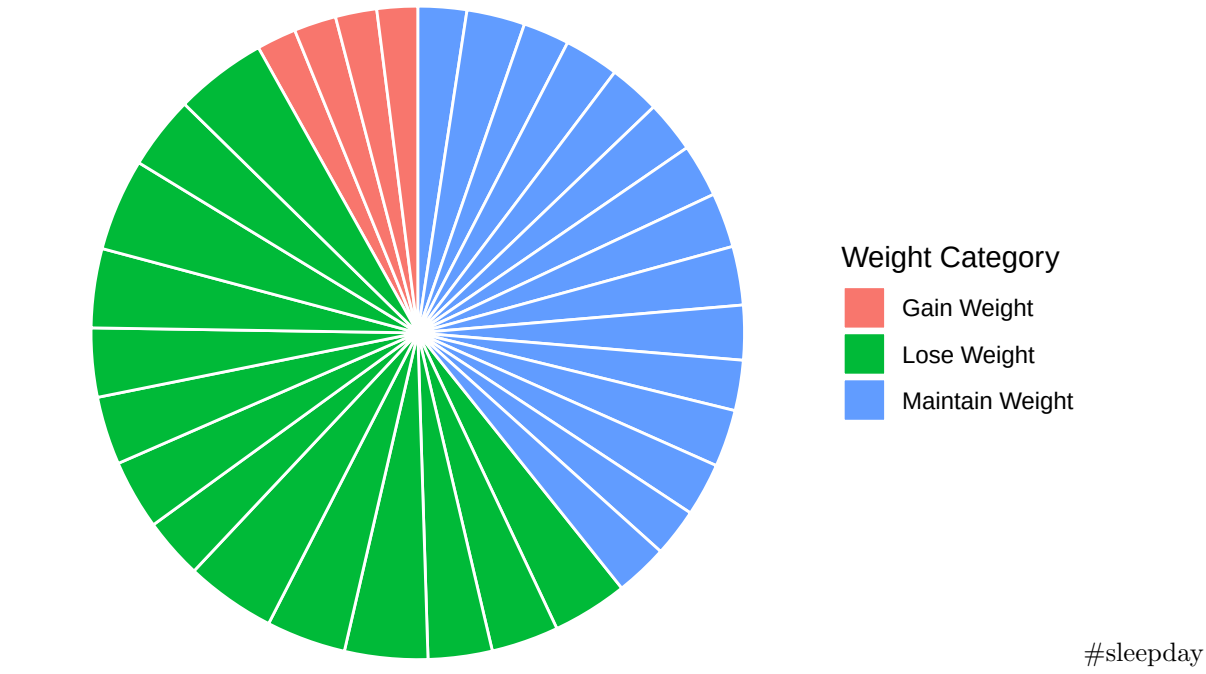
#visualizing the data

```
mean_values <- colMeans(daily_activity[, c("VeryActiveMinutes", "FairlyActiveMinutes", "LightlyActiveMi
#Create a data frame for ggplot
activity_data <- data.frame(
  Activity = c("Very Active", "Fairly Active", "Lightly Active", "Sedentary"),
  MeanMinutes = mean_values)

# Plot using ggplot with rotated X-axis labels
ggplot(activity_data, aes(x = reorder(Activity, MeanMinutes), y = MeanMinutes, fill = Activity)) +
  geom_col(color = "black") +
  labs(title = "Mean Activity Minutes", x = "Activity Type", y = "Average Minutes") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
avg_calories <- daily_activity %>%
  group_by(Id) %>%
  summarise(avg_Calories = mean(Calories))

# Categorize participants based on average calories
avg_calories <- mutate(avg_calories, WeightCategory = case_when(
```

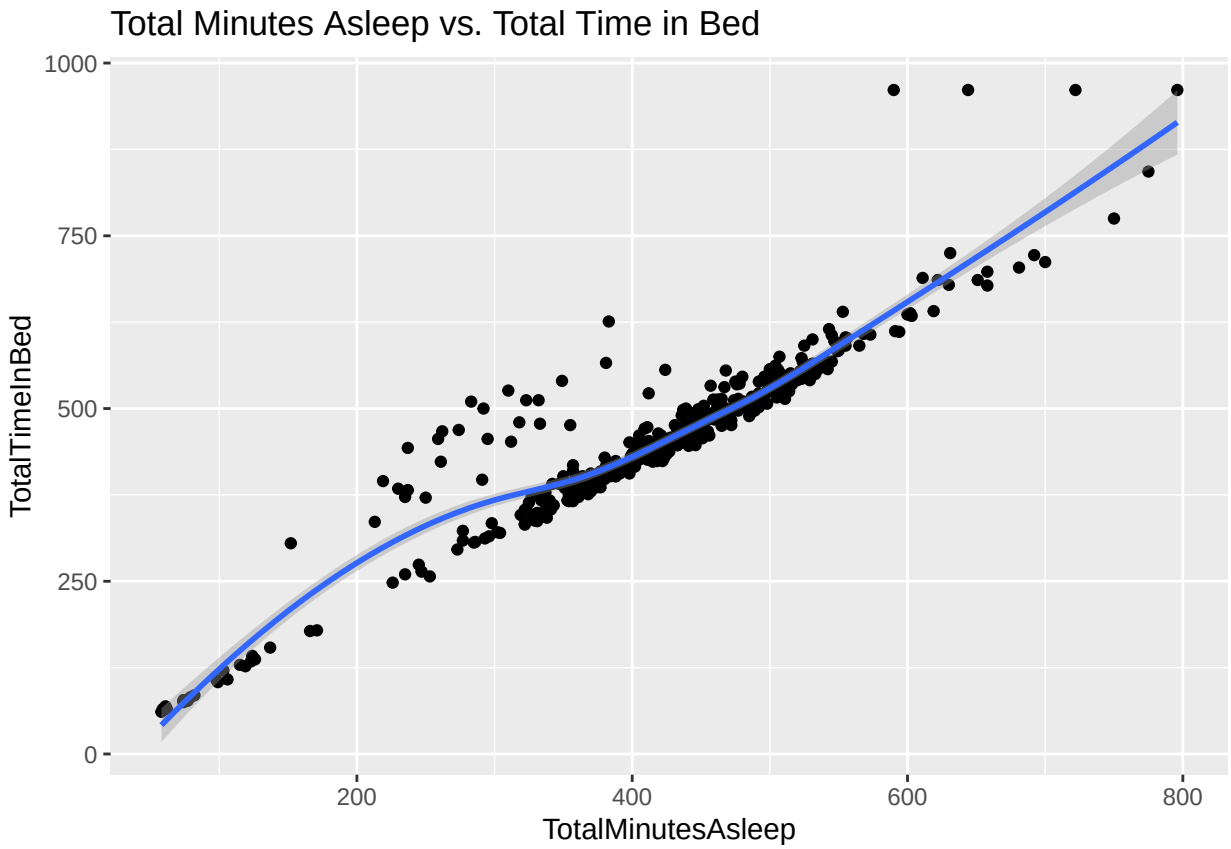
```
  avg_Calories >= 1600 & avg_Calories <= 2200 ~ 'Maintain Weight',
  avg_Calories < 1600 ~ 'Gain Weight',
  avg_Calories > 2200 ~ 'Lose Weight'
))

# Plot the bar chart
ggplot(avg_calories, aes(x = "", y = avg_Calories, fill = WeightCategory)) +
  geom_bar(stat = 'identity', width = 1, color = 'white') +
  coord_polar("y") +
  labs(title = 'Average Calories and Weight Category by Participant', fill = 'Weight Category') +
  theme_void()
```



#sleepday

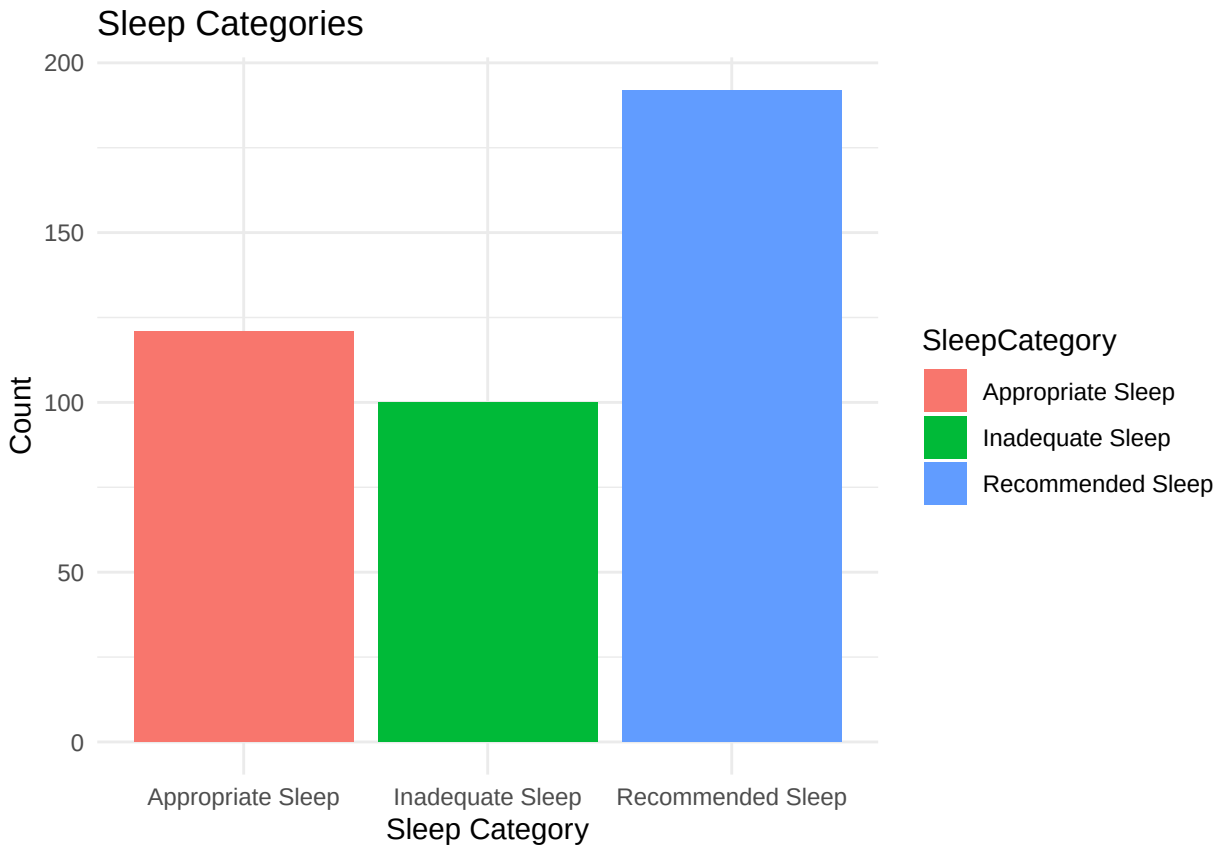
```
ggplot(sleepDay, aes(x = TotalMinutesAsleep, y = TotalTimeInBed)) +
  geom_point() +
  geom_smooth() +
  labs(title = "Total Minutes Asleep vs. Total Time in Bed")
```

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



it's intriguing to note the concentration of data points where individuals spent more time in bed than the actual duration of sleep. we can remind individuals to limit screen time before bedtime. Suggest establishing a calming bedtime routine to help individuals unwind before sleep. This may include activities such as reading, gentle stretching, or practicing relaxation techniques.

```
sleepDay <- mutate(sleepDay, SleepCategory = case_when(TotalMinutesAsleep >= 420 & TotalMinutesAsleep <=
  TRUE ~ 'Other' ))

ggplot(sleepDay, aes(x = SleepCategory, fill = SleepCategory)) + geom_bar() +labs(title = 'Sleep Categor
```




```
colnames(daily_activity)
```

```
##  [1] "Id"                       "ActivityDate"
##  [3] "TotalSteps"               "TotalDistance"
##  [5] "TrackerDistance"          "LoggedActivitiesDistance"
##  [7] "VeryActiveDistance"       "ModeratelyActiveDistance"
##  [9] "LightActiveDistance"      "SedentaryActiveDistance"
## [11] "VeryActiveMinutes"        "FairlyActiveMinutes"
## [13] "LightlyActiveMinutes"     "SedentaryMinutes"
## [15] "Calories"
```

#Merging the two datasets together

```
activity_sleep_combined <- merge(daily_activity, sleepDay, by = "Id", all = TRUE)
```

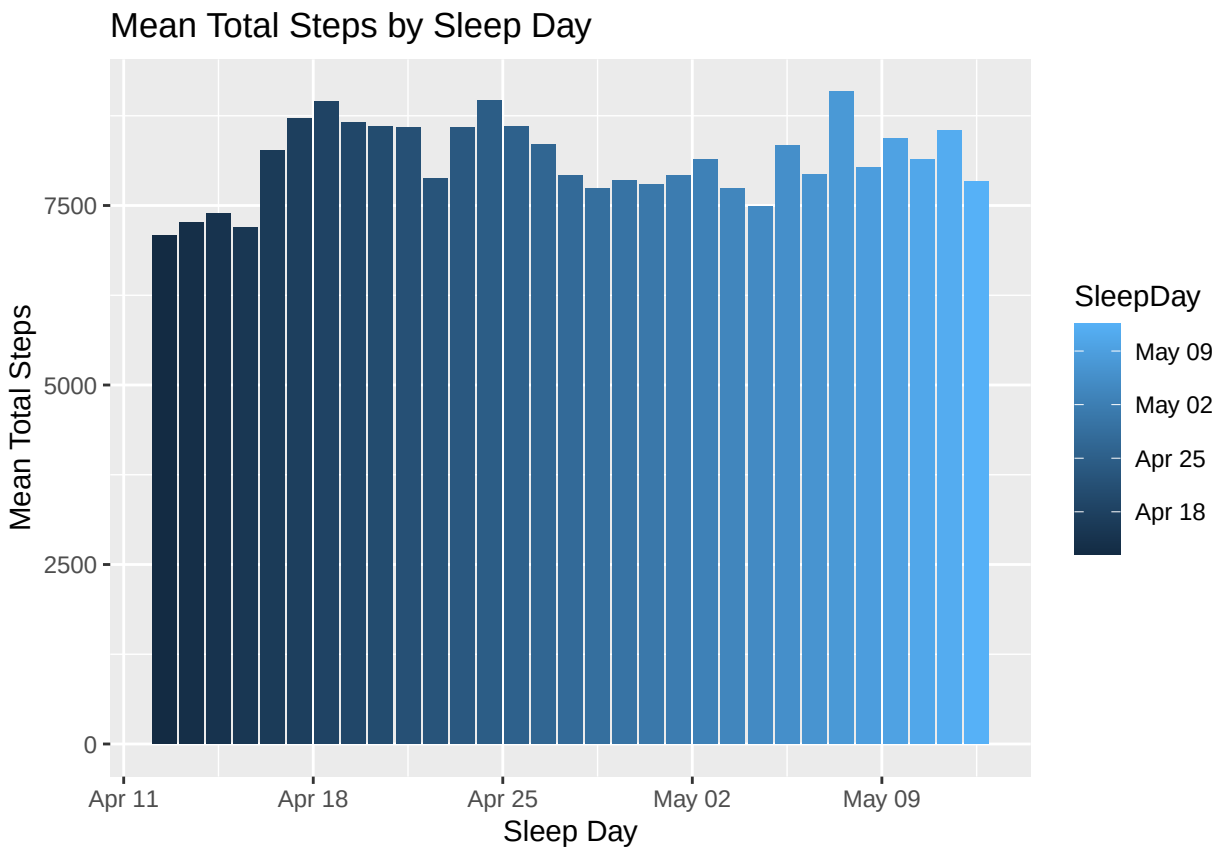
```
#View(activity_sleep_combined)
n_distinct(activity_sleep_combined$Id)
```

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

#Total Minutes Asleep vs. Total Steps

```
ggplot(data = activity_sleep_combined, aes(x = SleepDay, y = TotalSteps, fill = SleepDay)) +
  geom_bar(stat = "summary", fun = "mean") +
  xlab("Sleep Day") +
  ylab("Mean Total Steps") +
  ggtitle("Mean Total Steps by Sleep Day")
```

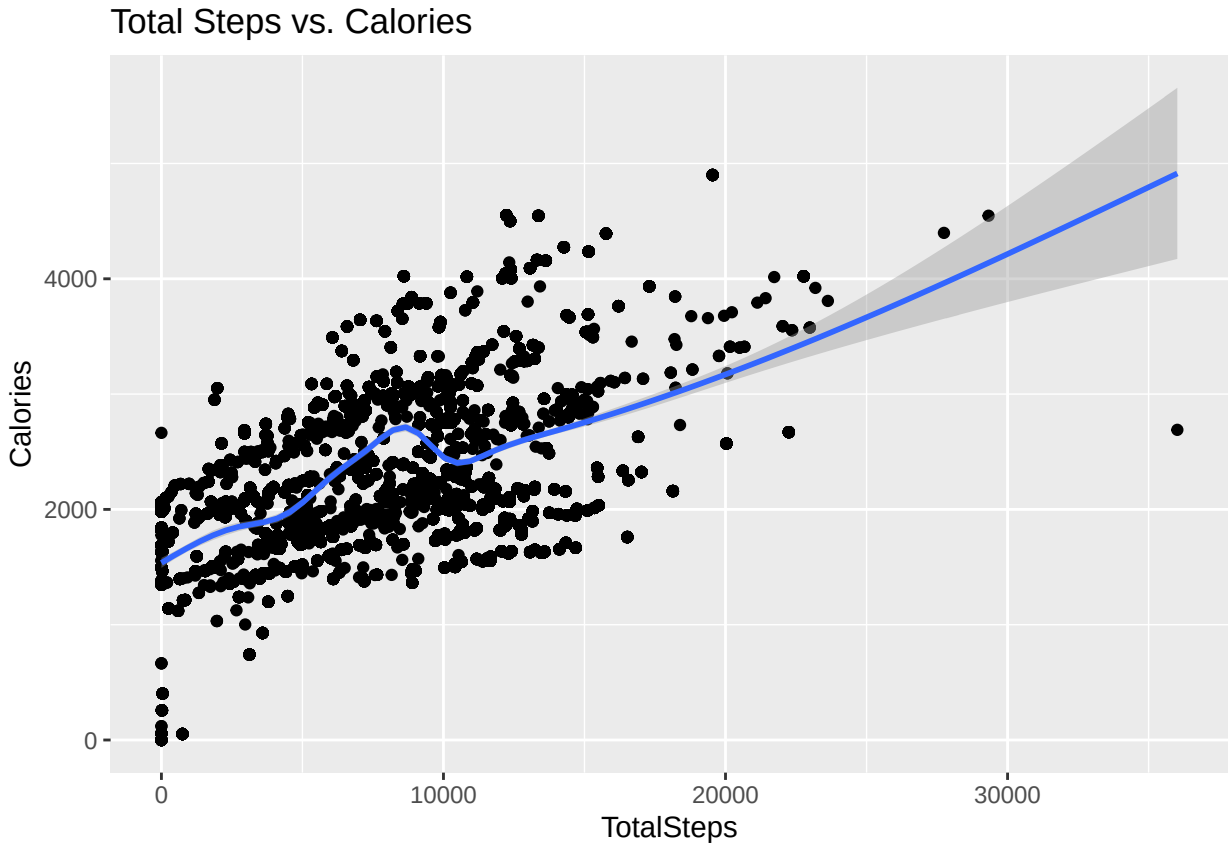
```
## Warning: Removed 227 rows containing non-finite values (`stat_summary()`).
```



there really is no relationship between sleeps and steps per day.

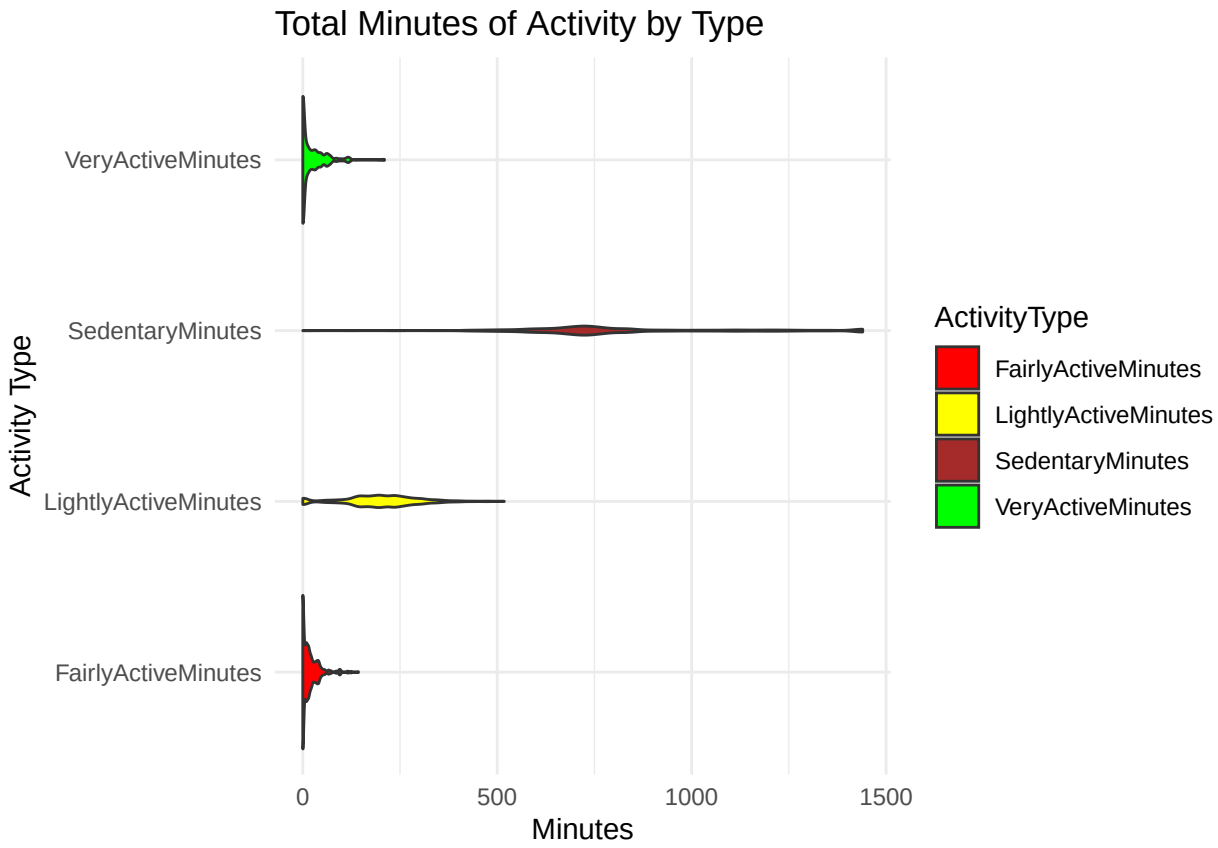
```
ggplot(data=activity_sleep_combined, aes(x=TotalSteps, y=Calories)) +
  geom_point() +
  geom_smooth() +
  labs(title="Total Steps vs. Calories")
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



the more steps a person takes, the more calories they burn. It's a great idea to remind users of that to reach their presumed goals.

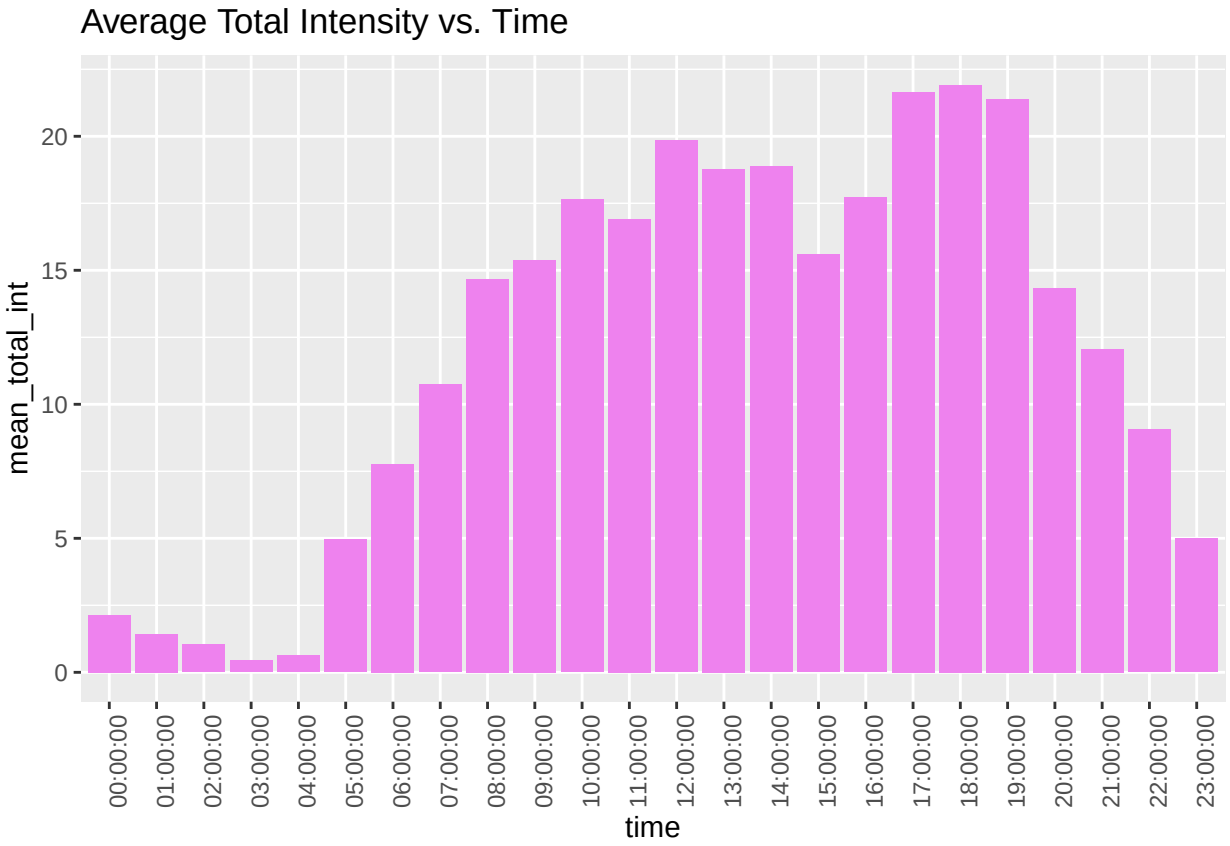
```
activity_data <- activity_sleep_combined %>%
  select(Id, VeryActiveMinutes, FairlyActiveMinutes,
         LightlyActiveMinutes, SedentaryMinutes) %>%
  gather(key = "ActivityType", value = "Minutes", -Id)
ggplot(activity_data, aes(x = ActivityType, y = Minutes, fill = ActivityType)) +
  geom_violin() +
  xlab("Activity Type") +
  ylab("Minutes") +
  ggtitle("Total Minutes of Activity by Type") +
  scale_fill_manual(values = c("red", "yellow", "brown", "green")) +
  theme_minimal() +
  coord_flip()
```



This visualization indicates that there is significant potential for promoting and encouraging physical activity.

```
int_new <- hourlyIntensities %>%
  group_by(time) %>%
  drop_na() %>%
  summarise(mean_total_int = mean(TotalIntensity))

ggplot(data=int_new, aes(x=time, y=mean_total_int)) + geom_histogram(stat = "identity", fill='violet') +
  theme(axis.text.x = element_text(angle = 90)) +
  labs(title="Average Total Intensity vs. Time")
```

```
## Warning in geom_histogram(stat = "identity", fill = "violet"): Ignoring unknown
## parameters: `binwidth`, `bins`, and `pad`
```



It's evident that individuals are most active between 5 pm and 7 pm, with additional peaks observed around 12 pm to 2 pm. These time intervals align with typical lunch breaks and the conclusion of the workday during the week. These periods present opportune moments to promote and encourage activities such as workouts, jogging, or taking a pet for a walk.

##Findings Analyzing the data from non-Bellabeat smart devices, we observe a prevalent pattern among women users who tend to engage in light activity, falling below the recommended daily walking levels. Encouraging these users to maintain healthier sleep and activity levels could be beneficial. The absence of weight tracking data suggests a potential disinterest or preference for privacy in this area.

The key takeaway for stakeholders is the importance of incorporating motivational alerts and educational notifications to drive increased user activity. The Bellabeat wellness watch offers a seamless solution, tracking activity, sleep, and stress without imposing daily log entries. Leveraging insights from peak activity hours, personalized prompts can effectively motivate women to enhance their activity levels.

Bellabeat's overarching mission to empower women aligns with the product's potential. Maximizing interactivity through alerts and notifications will not only engage users but also encourage them to explore and optimize their personal data. The focus should be on creating an app experience that actively supports women in unlocking their full potential.