Bellabeat case study

Sneha Pasalkar

2023-09-03

Start with installing packages and libraries

```
install.packages("janitor")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
install.packages("skimr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     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 error
```

```
library(readr)
library(ggplot2)
library(dplyr)
library(lubridate)
library(tidyr)
library(janitor)
```

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

```
library(skimr)
```

Now let us start importing files from dataset.

Please note : Data we are using is open source, available on `kaggle`

Please visit the link https://www.kaggle.com/datasets/arashnic/fitbit?resource=download to access data.

```
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.
```

```
calory <- read_csv("dailyCalories_merged.csv")
```

```
## Rows: 940 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): ActivityDay
## 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.
```

```
intensity <- read_csv("dailyIntensities_merged.csv")
```

```
## Rows: 940 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): ActivityDay
## dbl (9): Id, SedentaryMinutes, LightlyActiveMinutes, FairlyActiveMinutes, Ve...
##
## 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.
```

```
steps <- read_csv("dailySteps_merged.csv")
```

```
## Rows: 940 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): ActivityDay
## 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.
```

```
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.
```

```r
weight <- read_csv("weightLogInfo_merged.csv")
```

```
## Rows: 67 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Date
## dbl (6): Id, WeightKg, WeightPounds, Fat, BMI, LogId
## lgl (1): IsManualReport
##
## 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.
```

Now as all files have been fetched, let us check if those are loaded correctly, to know this will use some functions as below:

- **head()**: to check first few rows of data
- **colnames()** : to check column names from that dataframe.
- **str()** and **glimpse()** : to check summary of dataframe, we will also explore **skim.wihtout_chart()**

```r
head(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
colnames(activity)
```

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

```r
str(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>
skim_without_charts(activity)
```

Table 1: Data summary

Name	activity
Number of rows	940
Number of columns	15
Column type frequency:	
character	1
numeric	14
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ActivityDate	0	1	8	9	0	31	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	4.855407e+09	2.424805e+09	1.503960e+09	2.320127e+09	4.445115e+09	6.962181e+09	8.877689e+09
TotalSteps	0	1	7.637910e+03	5.087150e+03	0	3.789750e+03	7.405500e+03	1.072700e+04	3.601900e+04

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
TotalDistance	0	1	5.490000e+00	3.920000e+00	0	2.620000e+00	5.240000e+00	7.710000e+00	2.803000e+01
TrackerDistance	0	1	5.480000e+00	3.910000e+00	0	2.620000e+00	5.240000e+00	7.710000e+00	2.803000e+01
LoggedActivitiesDistance	0	1	1.100000e-01	6.200000e-01	0	0.000000e+00	0.000000e+00	0.000000e+00	4.940000e+00
VeryActiveDistance	0	1	1.500000e+00	2.066000e+00	0	0.000000e+00	2.000000e-01	2.050000e+00	2.092000e+01
ModeratelyActiveDistance	0	1	5.700000e-01	8.800000e-01	0	0.000000e+00	2.000000e-01	8.000000e-01	6.480000e+00
LightActiveDistance	0	1	3.340000e+00	2.040000e+00	0	1.950000e+00	3.060000e+00	4.780000e+00	1.071000e+01
SedentaryActiveDistance	0	1	0.000000e+00	1.000000e-02	0	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e-01
VeryActiveMinutes	0	1	2.116000e+01	3.284000e+01	0	0.000000e+00	4.000000e+00	3.200000e+01	2.100000e+02
FairlyActiveMinutes	0	1	1.356000e+01	1.999000e+01	0	0.000000e+00	6.000000e+00	1.900000e+01	1.043000e+02
LightlyActiveMinutes	0	1	1.928100e+02	1.091700e+02	0	1.270000e+02	1.929000e+02	2.640000e+02	5.028000e+02
SedentaryMinutes	0	1	9.912100e+02	3.012700e+02	0	7.297500e+02	1.057500e+03	1.329500e+03	1.734000e+03
Calories	0	1	2.303610e+03	7.181700e+02	0	1.828500e+03	2.134000e+03	2.793250e+03	4.900000e+03

Using this we checked if our columns have correct data type or not?

Here we observed below: * column - ActivityDate is showing character data type, wherein it should have date datatype * Naming conventions used in this data frame for column names is combination of Upper case and lower case letters, we will use ~**clean_names()**‘ to solve this as part of our data cleaning process.

Before starting cleaning process, let us go through all data frames we have loaded to check how much cleaning is required before starting with analysis phase of the data.

We will stick to `head()`, `colnames()` and `str()` to achieve this.

```
head(calory)
```

```
## # A tibble: 6 x 3
##           Id ActivityDay Calories
##        <dbl> <chr>          <dbl>
## 1 1503960366 4/12/2016       1985
## 2 1503960366 4/13/2016       1797
## 3 1503960366 4/14/2016       1776
## 4 1503960366 4/15/2016       1745
## 5 1503960366 4/16/2016       1863
## 6 1503960366 4/17/2016       1728
```

```
colnames(calory)
```

```
## [1] "Id"          "ActivityDay" "Calories"
```

```
str(calory)
```

```
## spc_tbl_ [940 x 3] (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 ...
##  $ ActivityDay: chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ Calories   : num [1:940] 1985 1797 1776 1745 1863 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   ActivityDay = col_character(),
##   ..   Calories = col_double()
```

```
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
head(intensity)
```

```
## # A tibble: 6 x 10
##         Id ActivityDay SedentaryMinutes LightlyActiveMinutes FairlyActiveMinutes
##      <dbl> <chr>                  <dbl>                <dbl>                <dbl>
## 1   1.50e9 4/12/2016                728                  328                   13
## 2   1.50e9 4/13/2016                776                  217                   19
## 3   1.50e9 4/14/2016               1218                  181                   11
## 4   1.50e9 4/15/2016                726                  209                   34
## 5   1.50e9 4/16/2016                773                  221                   10
## 6   1.50e9 4/17/2016                539                  164                   20
## # i 5 more variables: VeryActiveMinutes <dbl>, SedentaryActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, moderatelyActiveDistance <dbl>,
## #   VeryActiveDistance <dbl>
```

```
colnames(intensity)
```

```
##  [1] "Id"                       "ActivityDay"
##  [3] "SedentaryMinutes"         "LightlyActiveMinutes"
##  [5] "FairlyActiveMinutes"      "VeryActiveMinutes"
##  [7] "SedentaryActiveDistance"  "LightActiveDistance"
##  [9] "moderatelyActiveDistance" "VeryActiveDistance"
```

```
str(intensity)
```

```
## spc_tbl_ [940 x 10] (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 ...
##  $ ActivityDay              : chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ SedentaryMinutes         : num [1:940] 728 776 1218 726 773 ...
##  $ LightlyActiveMinutes     : num [1:940] 328 217 181 209 221 164 233 264 205 211 ...
##  $ FairlyActiveMinutes      : num [1:940] 13 19 11 34 10 20 16 31 12 8 ...
##  $ VeryActiveMinutes        : num [1:940] 25 21 30 29 36 38 42 50 28 19 ...
##  $ SedentaryActiveDistance : num [1:940] 0 0 0 0 0 0 0 0 0 0 ...
##  $ LightActiveDistance      : num [1:940] 6.06 4.71 3.91 2.83 5.04 ...
##  $ moderatelyActiveDistance: num [1:940] 0.55 0.69 0.4 1.26 0.41 ...
##  $ VeryActiveDistance       : num [1:940] 1.88 1.57 2.44 2.14 2.71 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   ActivityDay = col_character(),
##   ..   SedentaryMinutes = col_double(),
##   ..   LightlyActiveMinutes = col_double(),
##   ..   FairlyActiveMinutes = col_double(),
##   ..   VeryActiveMinutes = col_double(),
##   ..   SedentaryActiveDistance = col_double(),
##   ..   LightActiveDistance = col_double(),
##   ..   moderatelyActiveDistance = col_double(),
##   ..   VeryActiveDistance = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
head(steps)
```

```
## # A tibble: 6 x 3
```

```
##           Id ActivityDay StepTotal
##        <dbl> <chr>           <dbl>
## 1 1503960366 4/12/2016       13162
## 2 1503960366 4/13/2016       10735
## 3 1503960366 4/14/2016       10460
## 4 1503960366 4/15/2016        9762
## 5 1503960366 4/16/2016       12669
## 6 1503960366 4/17/2016        9705
```

```
colnames(steps)
```

```
## [1] "Id"          "ActivityDay" "StepTotal"
```

```
str(steps)
```

```
## spc_tbl_ [940 x 3] (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 ...
##  $ ActivityDay: chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ StepTotal  : num [1:940] 13162 10735 10460 9762 12669 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   ActivityDay = col_character(),
##   ..   StepTotal = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
head(sleep)
```

```
## # A tibble: 6 x 5
##           Id SleepDay        TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##        <dbl> <chr>                       <dbl>              <dbl>          <dbl>
## 1 1503960366 4/12/2016 12:0~                1                327            346
## 2 1503960366 4/13/2016 12:0~                2                384            407
## 3 1503960366 4/15/2016 12:0~                1                412            442
## 4 1503960366 4/16/2016 12:0~                2                340            367
## 5 1503960366 4/17/2016 12:0~                1                700            712
## 6 1503960366 4/19/2016 12:0~                1                304            320
```

```
colnames(sleep)
```

```
## [1] "Id"                 "SleepDay"           "TotalSleepRecords"
## [4] "TotalMinutesAsleep" "TotalTimeInBed"
```

```
str(sleep)
```

```
## spc_tbl_ [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" "4/15/2016 12:00:
##  $ 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 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   SleepDay = col_character(),
##   ..   TotalSleepRecords = col_double(),
##   ..   TotalMinutesAsleep = col_double(),
```

```
##   ..   TotalTimeInBed = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
head(weight)
```

```
## # A tibble: 6 x 8
##           Id Date       WeightKg WeightPounds   Fat   BMI IsManualReport   LogId
##        <dbl> <chr>         <dbl>        <dbl> <dbl> <dbl> <lgl>            <dbl>
## 1 1503960366 5/2/2016 ~     52.6         116.    22  22.6 TRUE           1.46e12
## 2 1503960366 5/3/2016 ~     52.6         116.    NA  22.6 TRUE           1.46e12
## 3 1927972279 4/13/2016~    134.          294.    NA  47.5 FALSE          1.46e12
## 4 2873212765 4/21/2016~     56.7         125.    NA  21.5 TRUE           1.46e12
## 5 2873212765 5/12/2016~     57.3         126.    NA  21.7 TRUE           1.46e12
## 6 4319703577 4/17/2016~     72.4         160.    25  27.5 TRUE           1.46e12
```

```
colnames(weight)
```

```
## [1] "Id"             "Date"           "WeightKg"       "WeightPounds"
## [5] "Fat"            "BMI"            "IsManualReport" "LogId"
```

```
str(weight)
```

```
## spc_tbl_ [67 x 8] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Id            : num [1:67] 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...
##  $ Date          : chr [1:67] "5/2/2016 11:59:59 PM" "5/3/2016 11:59:59 PM" "4/13/2016 1:08:52 AM" "
##  $ WeightKg      : num [1:67] 52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num [1:67] 116 116 294 125 126 ...
##  $ Fat           : num [1:67] 22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num [1:67] 22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: logi [1:67] TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num [1:67] 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   Date = col_character(),
##   ..   WeightKg = col_double(),
##   ..   WeightPounds = col_double(),
##   ..   Fat = col_double(),
##   ..   BMI = col_double(),
##   ..   IsManualReport = col_logical(),
##   ..   LogId = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

We observed below details:

- calory : Only 3 columns, Datatype of ActivityDay is incorrect.
- intensity : total 9 columns, Datatype of ActivityDay is incorrect
- steps: total 3 columns, Datatype of ActivityDay is incorrect
- sleep: total 5 colums, Datatype of sleepDay is incorrect, we need to set it as date and time instead of character, we can also rename it to match other dataframes.
- weight : total 8 columns, Datatype of Date is incorrect

Based on our observations lets clean our data

- First we will start by activity dataframe, will change name of column ActivityDate to Date and change the data type from character to Date format

```
activity_1 <- activity %>%
  rename (date = ActivityDate)
```

Now lets check date in activity dataframe

```
head(activity_1)
```

```
## # A tibble: 6 x 15
##         Id date  TotalSteps TotalDistance TrackerDistance LoggedActivitiesDist~1
##      <dbl> <chr>      <dbl>         <dbl>           <dbl>                   <dbl>
## 1   1.50e9 4/12~      13162          8.5             8.5                        0
## 2   1.50e9 4/13~      10735          6.97            6.97                       0
## 3   1.50e9 4/14~      10460          6.74            6.74                       0
## 4   1.50e9 4/15~       9762          6.28            6.28                       0
## 5   1.50e9 4/16~      12669          8.16            8.16                       0
## 6   1.50e9 4/17~       9705          6.48            6.48                       0
## # i abbreviated name: 1: LoggedActivitiesDistance
## # i 9 more variables: VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

```
str(activity_1)
```

```
## 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 ...
##  $ date                     : 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>
```

We have renamed the column now let us change it to date format, to do this will first check existing date to change format accordingly.

```
head(activity_1)
```

```
## # A tibble: 6 x 15
##         Id date  TotalSteps TotalDistance TrackerDistance LoggedActivitiesDist~1
##      <dbl> <chr>      <dbl>         <dbl>           <dbl>                   <dbl>
## 1   1.50e9 4/12~      13162          8.5             8.5                        0
## 2   1.50e9 4/13~      10735          6.97            6.97                       0
## 3   1.50e9 4/14~      10460          6.74            6.74                       0
## 4   1.50e9 4/15~       9762          6.28            6.28                       0
## 5   1.50e9 4/16~      12669          8.16            8.16                       0
## 6   1.50e9 4/17~       9705          6.48            6.48                       0
## # i abbreviated name: 1: LoggedActivitiesDistance
## # i 9 more variables: VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

We observed that we have date in mmddyy format hence will use **mdy()** to change it in correct format.

```
activity_1$date <- mdy(activity_1$date)
head(activity_1)
```

```
## # A tibble: 6 x 15
##           Id date       TotalSteps TotalDistance TrackerDistance
##        <dbl> <date>          <dbl>         <dbl>           <dbl>
## 1 1503960366 2016-04-12      13162          8.5             8.5
## 2 1503960366 2016-04-13      10735          6.97            6.97
## 3 1503960366 2016-04-14      10460          6.74            6.74
## 4 1503960366 2016-04-15       9762          6.28            6.28
## 5 1503960366 2016-04-16      12669          8.16            8.16
## 6 1503960366 2016-04-17       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>
```

Now Let us perform same for data frames: Calory, intensity, steps.

We are creating new data frames as it will help us fetch original once if anything goes wrong while cleaning and organizing data.

```
calory_1 <- calory %>%
  rename (date = ActivityDay)

# changing format
calory_1$date <- mdy(calory_1$date)
```

```r
head(calory_1)
```

```
## # A tibble: 6 x 3
##           Id date       Calories
##        <dbl> <date>        <dbl>
## 1 1503960366 2016-04-12     1985
## 2 1503960366 2016-04-13     1797
## 3 1503960366 2016-04-14     1776
## 4 1503960366 2016-04-15     1745
## 5 1503960366 2016-04-16     1863
## 6 1503960366 2016-04-17     1728
```

```r
#intensity data frame
intensity_1 <- intensity %>%
  rename (date = ActivityDay)

# changing format
intensity_1$date <- mdy(intensity_1$date)
head(intensity_1)
```

```
## # A tibble: 6 x 10
##           Id date       SedentaryMinutes LightlyActiveMinutes FairlyActiveMinutes
##        <dbl> <date>                <dbl>                <dbl>                <dbl>
## 1    1.50e9 2016-04-12              728                  328                   13
## 2    1.50e9 2016-04-13              776                  217                   19
## 3    1.50e9 2016-04-14             1218                  181                   11
## 4    1.50e9 2016-04-15              726                  209                   34
## 5    1.50e9 2016-04-16              773                  221                   10
## 6    1.50e9 2016-04-17              539                  164                   20
## # i 5 more variables: VeryActiveMinutes <dbl>, SedentaryActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, moderatelyActiveDistance <dbl>,
## #   VeryActiveDistance <dbl>
```

```r
#steps data frame
steps_1 <- steps %>%
  rename (date = ActivityDay)

#changing format
steps_1$date <- mdy(steps_1$date)
head(steps_1)
```

```
## # A tibble: 6 x 3
##           Id date       StepTotal
##        <dbl> <date>         <dbl>
## 1 1503960366 2016-04-12     13162
## 2 1503960366 2016-04-13     10735
## 3 1503960366 2016-04-14     10460
## 4 1503960366 2016-04-15      9762
## 5 1503960366 2016-04-16     12669
## 6 1503960366 2016-04-17      9705
```

For data frames sleep and weight we have date as character for but we have time as well along with it. For these two data frames will change the name and format to date plus will create two separate columns which should display date in one column and time in other.

```r
sleep_1 <- sleep %>%
  rename (date = SleepDay)

#changing format
head(sleep_1)
```

```
## # A tibble: 6 x 5
##           Id date            TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##        <dbl> <chr>                       <dbl>               <dbl>          <dbl>
## 1 1503960366 4/12/2016 12:0~                 1                 327            346
## 2 1503960366 4/13/2016 12:0~                 2                 384            407
## 3 1503960366 4/15/2016 12:0~                 1                 412            442
## 4 1503960366 4/16/2016 12:0~                 2                 340            367
## 5 1503960366 4/17/2016 12:0~                 1                 700            712
## 6 1503960366 4/19/2016 12:0~                 1                 304            320
```

```r
str(sleep_1)
```

```
## spc_tbl_ [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 ...
##  $ date              : chr [1:413] "4/12/2016 12:00:00 AM" "4/13/2016 12:00:00 AM" "4/15/2016 12:00:
##  $ 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 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Id = col_double(),
##   ..   SleepDay = col_character(),
##   ..   TotalSleepRecords = col_double(),
##   ..   TotalMinutesAsleep = col_double(),
##   ..   TotalTimeInBed = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```r
sleep_2 <- separate(sleep_1, date, into = c("date", "time"), sep = ' ')
```

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

```r
head(sleep_2)
```

```
## # A tibble: 6 x 6
##           Id date      time  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##        <dbl> <chr>     <chr>             <dbl>               <dbl>          <dbl>
## 1 1503960366 4/12/2016 12:0~                 1                 327            346
## 2 1503960366 4/13/2016 12:0~                 2                 384            407
## 3 1503960366 4/15/2016 12:0~                 1                 412            442
## 4 1503960366 4/16/2016 12:0~                 2                 340            367
## 5 1503960366 4/17/2016 12:0~                 1                 700            712
## 6 1503960366 4/19/2016 12:0~                 1                 304            320
```

```r
str(sleep_2)
```

```
## tibble [413 x 6] (S3: tbl_df/tbl/data.frame)
##  $ Id                : num [1:413] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ date              : chr [1:413] "4/12/2016" "4/13/2016" "4/15/2016" "4/16/2016" ...
##  $ time              : chr [1:413] "12:00:00" "12:00:00" "12:00:00" "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 ...
```

```
sleep_2$date <- mdy(sleep_2$date)

str(sleep_2)
```

```
## tibble [413 x 6] (S3: tbl_df/tbl/data.frame)
##  $ Id                : num [1:413] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ date              : Date[1:413], format: "2016-04-12" "2016-04-13" ...
##  $ time              : chr [1:413] "12:00:00" "12:00:00" "12:00:00" "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 ...
```

```
head(sleep_2)
```

```
## # A tibble: 6 x 6
##          Id date       time  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##       <dbl> <date>     <chr>             <dbl>              <dbl>          <dbl>
## 1    1.50e9 2016-04-12 12:0~                 1                327            346
## 2    1.50e9 2016-04-13 12:0~                 2                384            407
## 3    1.50e9 2016-04-15 12:0~                 1                412            442
## 4    1.50e9 2016-04-16 12:0~                 2                340            367
## 5    1.50e9 2016-04-17 12:0~                 1                700            712
## 6    1.50e9 2016-04-19 12:0~                 1                304            320
```

Now will change for weight column.

```
sleep_2$time <- hms(sleep_2$time)
str(sleep_2)
```

```
## tibble [413 x 6] (S3: tbl_df/tbl/data.frame)
##  $ Id                : num [1:413] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ date              : Date[1:413], format: "2016-04-12" "2016-04-13" ...
##  $ time              :Formal class 'Period' [package "lubridate"] with 6 slots
##   .. ..@ .Data : num [1:413] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ year  : num [1:413] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ month : num [1:413] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ day   : num [1:413] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ hour  : num [1:413] 12 12 12 12 12 12 12 12 12 12 ...
##   .. ..@ minute: num [1:413] 0 0 0 0 0 0 0 0 0 0 ...
##  $ 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 ...
```

```
head(sleep_2)
```

```
## # A tibble: 6 x 6
##           Id date       time      TotalSleepRecords TotalMinutesAsleep
##        <dbl> <date>     <Period>              <dbl>              <dbl>
## 1 1503960366 2016-04-12 12H 0M 0S                 1                327
## 2 1503960366 2016-04-13 12H 0M 0S                 2                384
## 3 1503960366 2016-04-15 12H 0M 0S                 1                412
## 4 1503960366 2016-04-16 12H 0M 0S                 2                340
## 5 1503960366 2016-04-17 12H 0M 0S                 1                700
```

```
## 6 1503960366 2016-04-19 12H 0M 0S                 1                304
## # i 1 more variable: TotalTimeInBed <dbl>
```

```
weight_1 <- separate(weight, Date, into = c("date", "time"), sep = ' ')
```

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

```
weight_1$date <- mdy(weight_1$date)
head(weight_1)
```

```
## # A tibble: 6 x 9
##           Id date       time    WeightKg WeightPounds   Fat   BMI IsManualReport
##        <dbl> <date>     <chr>      <dbl>        <dbl> <dbl> <dbl> <lgl>
## 1 1503960366 2016-05-02 11:59:~     52.6         116.    22  22.6 TRUE
## 2 1503960366 2016-05-03 11:59:~     52.6         116.    NA  22.6 TRUE
## 3 1927972279 2016-04-13 1:08:52    134.          294.    NA  47.5 FALSE
## 4 2873212765 2016-04-21 11:59:~     56.7         125.    NA  21.5 TRUE
## 5 2873212765 2016-05-12 11:59:~     57.3         126.    NA  21.7 TRUE
## 6 4319703577 2016-04-17 11:59:~     72.4         160.    25  27.5 TRUE
## # i 1 more variable: LogId <dbl>
```

```
str(weight_1)
```

```
## tibble [67 x 9] (S3: tbl_df/tbl/data.frame)
##  $ Id            : num [1:67] 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...
##  $ date          : Date[1:67], format: "2016-05-02" "2016-05-03" ...
##  $ time          : chr [1:67] "11:59:59" "11:59:59" "1:08:52" "11:59:59" ...
##  $ WeightKg      : num [1:67] 52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num [1:67] 116 116 294 125 126 ...
##  $ Fat           : num [1:67] 22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num [1:67] 22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: logi [1:67] TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num [1:67] 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...
```

```
weight_1$time <- hms(weight_1$time)
str(weight_1)
```

```
## tibble [67 x 9] (S3: tbl_df/tbl/data.frame)
##  $ Id            : num [1:67] 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...
##  $ date          : Date[1:67], format: "2016-05-02" "2016-05-03" ...
##  $ time          :Formal class 'Period' [package "lubridate"] with 6 slots
##   .. ..@ .Data : num [1:67] 59 59 52 59 59 59 59 59 59 59 ...
##   .. ..@ year  : num [1:67] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ month : num [1:67] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ day   : num [1:67] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..@ hour  : num [1:67] 11 11 1 11 11 11 11 11 11 11 ...
##   .. ..@ minute: num [1:67] 59 59 8 59 59 59 59 59 59 59 ...
##  $ WeightKg      : num [1:67] 52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num [1:67] 116 116 294 125 126 ...
##  $ Fat           : num [1:67] 22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num [1:67] 22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: logi [1:67] TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num [1:67] 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...
```

```
head(weight_1)
```

```
## # A tibble: 6 x 9
```

```
##       Id date       time        WeightKg WeightPounds   Fat   BMI IsManualReport
##    <dbl> <date>     <Period>       <dbl>        <dbl> <dbl> <dbl> <lgl>
## 1 1.50e9 2016-05-02 11H 59M 59S     52.6         116.    22  22.6 TRUE
## 2 1.50e9 2016-05-03 11H 59M 59S     52.6         116.    NA  22.6 TRUE
## 3 1.93e9 2016-04-13 1H 8M 52S      134.          294.    NA  47.5 FALSE
## 4 2.87e9 2016-04-21 11H 59M 59S     56.7         125.    NA  21.5 TRUE
## 5 2.87e9 2016-05-12 11H 59M 59S     57.3         126.    NA  21.7 TRUE
## 6 4.32e9 2016-04-17 11H 59M 59S     72.4         160.    25  27.5 TRUE
## # i 1 more variable: LogId <dbl>
```

Now we have all data frames with correct data type.

To move ahead we will make sure we have consitent column names. To achieve this consistency will use **clean_names()**

```
activity_1 <- clean_names(activity_1)
colnames(activity_1)
```

```
##  [1] "id"                          "date"
##  [3] "total_steps"                 "total_distance"
##  [5] "tracker_distance"            "logged_activities_distance"
##  [7] "very_active_distance"        "moderately_active_distance"
##  [9] "light_active_distance"       "sedentary_active_distance"
## [11] "very_active_minutes"         "fairly_active_minutes"
## [13] "lightly_active_minutes"      "sedentary_minutes"
## [15] "calories"
```

We observed that all column names changed to all small case letters with _ in between them.

Lets do the same for remaining data frames: **Calory_1, intensity_1, sleep_2, steps_1, weight_1**

```
calory_1 <- clean_names(calory_1)
colnames(calory_1)
```

```
## [1] "id"       "date"     "calories"
```

```
intensity_1 <- clean_names(intensity_1)
colnames(intensity_1)
```

```
##  [1] "id"                          "date"
##  [3] "sedentary_minutes"           "lightly_active_minutes"
##  [5] "fairly_active_minutes"       "very_active_minutes"
##  [7] "sedentary_active_distance"   "light_active_distance"
##  [9] "moderately_active_distance" "very_active_distance"
```

```
sleep_2 <- clean_names(sleep_2)
colnames(sleep_2)
```

```
## [1] "id"                   "date"                 "time"
## [4] "total_sleep_records"  "total_minutes_asleep" "total_time_in_bed"
```

```
steps_1 <- clean_names(steps_1)
colnames(steps_1)
```

```
## [1] "id"         "date"       "step_total"
```

```
weight_1 <- clean_names(weight_1)
colnames(weight_1)
```

```
## [1] "id"               "date"             "time"             "weight_kg"
```

```
## [5] "weight_pounds"    "fat"              "bmi"              "is_manual_report"
## [9] "log_id"
```

Now let us check for unique valus.

```
n_distinct(activity_1$id)
```

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

```
n_distinct(calory_1$id)
```

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

```
n_distinct(intensity_1$id)
```

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

```
n_distinct(steps_1$id)
```

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

```
n_distinct(sleep_2$id)
```

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

```
n_distinct(weight_1$id)
```

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

Lets check if we have duplicates now.

```
sum(duplicated(activity_1))
```

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

```
sum(duplicated(calory_1))
```

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

```
sum(duplicated(intensity_1))
```

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

```
sum(duplicated(sleep_2))
```

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

```
sum(duplicated(steps_1))
```

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

```
sum(duplicated(weight_1))
```

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

Here we observed that we have 3 duplicates in Sleep data frame, will drop them using **drop()**

```
sleep_2<- sleep_2 %>%
  distinct() %>%
  drop_na()

# check if duplicates are droped
sum(duplicated(sleep_2))
```

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

Finally we have clean our data and **Prepare** phase of data analysis is now completed.

We are now ready to move further for our **Analysis** phase.

Here,

- will do some analysis with data available in each data frame
- will merge two data frames for our analysis
- will make some categorizations to analyze data

Lets get started:

```
activity_1 %>% select(total_steps, total_distance, tracker_distance, sedentary_minutes, calories) %>%
  summary()
```

```
##   total_steps    total_distance   tracker_distance sedentary_minutes
##  Min.   :    0   Min.   : 0.000   Min.   : 0.000   Min.   :   0.0
##  1st Qu.: 3790   1st Qu.: 2.620   1st Qu.: 2.620   1st Qu.: 729.8
##  Median : 7406   Median : 5.245   Median : 5.245   Median :1057.5
##  Mean   : 7638   Mean   : 5.490   Mean   : 5.475   Mean   : 991.2
##  3rd Qu.:10727   3rd Qu.: 7.713   3rd Qu.: 7.710   3rd Qu.:1229.5
##  Max.   :36019   Max.   :28.030   Max.   :28.030   Max.   :1440.0
##     calories
##  Min.   :   0
##  1st Qu.:1828
##  Median :2134
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900
```

Above data shows average steps taken by individual is 7406, which is quite low, as per research average steps for a healthy adult shoud be 10000.

- Based on this we can suggest bellabeat to make strategy to motivate users for 10000 steps
- client can start showing reminder to complete 10000 steps

Data shows on an average user covers a distance of 5 km which is good sign

- client can opt in for special customized alerts based on users previous data, if user is walking for 5 km per day generally, app should notify them if it is not done on specific day.

The data also shows sedentary minutes average as 1057.5, which is too much and which is not good if being active is the goal of user.

In this case there are three scenarios:

- When user is inactive due to working style (working on system) then it may capture that time as sedentary as movement is less.
- When user is asleep for some time
- When user is travelling

As we have limited data for user and we do not have much details as when this sedentary time was captured we can not reach out to root cause of this, however bellabeat can start having notifications if idle time is more than an hour to encourage physical activity

- Sending notifications for idle time and allowing user to snooze it or to set it for customized time can be done.

Our data shows average calories burned are 2134. According to the Dietary Guidelines for Americans 2020–2025, the average adult woman burns roughly 1,600 to 2,400 calories per day so our data matches with the expectations.

- Bellabeat can use sending notifications here too to motivate users to progress on a fitness path

```
activity_1 %>% select(very_active_minutes,fairly_active_minutes, lightly_active_minutes) %>%
  summary()
```

```
##  very_active_minutes fairly_active_minutes lightly_active_minutes
##  Min.   :  0.00      Min.   :  0.00        Min.   :  0.0
##  1st Qu.:  0.00      1st Qu.:  0.00        1st Qu.:127.0
##  Median :  4.00      Median :  6.00        Median :199.0
##  Mean   : 21.16      Mean   : 13.56        Mean   :192.8
##  3rd Qu.: 32.00      3rd Qu.: 19.00        3rd Qu.:264.0
##  Max.   :210.00      Max.   :143.00        Max.   :518.0
```

```
calory_1 %>% select(calories) %>%
  summary()
```

```
##     calories
##  Min.   :   0
##  1st Qu.:1828
##  Median :2134
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900
```

```
sleep_2 %>% select(total_sleep_records,total_minutes_asleep,total_time_in_bed) %>%
  summary()
```

```
##  total_sleep_records total_minutes_asleep total_time_in_bed
##  Min.   :1.00        Min.   : 58.0        Min.   : 61.0
##  1st Qu.:1.00        1st Qu.:361.0        1st Qu.:403.8
##  Median :1.00        Median :432.5        Median :463.0
##  Mean   :1.12        Mean   :419.2        Mean   :458.5
##  3rd Qu.:1.00        3rd Qu.:490.0        3rd Qu.:526.0
##  Max.   :3.00        Max.   :796.0        Max.   :961.0
```

```
weight_1 %>% select(weight_kg,bmi) %>%
  summary()
```

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

Based on above observations we can conclude: * Users are mostly active for 4 minutes only * Users are fairly active for 6 minutes only * Mostly users are lightly active * Average user sleeps for 7 hours * Average time in bed is also around 7.64 hours

Here our data has some limitations as we dont know sleep category, there are some apps which shows subcategory in sleep as deep sleep, light sleep

- bellabeat can use such subcategory to provide more customized user experience
- bellabeat can work on creating a function which alows user to schedule sleep routine
- bellabeat can also work on sending notifications when sleep routine is about to start, so that user can work on their sleeping patterns as well, because study shows that 8 hours of sleep is required for a

healthy life.

- Data shows average weight of user is 62.5 KG
- Average BMI of user is 25.19, this is not good as BMI > 24.9 is considered as obese
- bellabeat can work on suggesting low calorie diets or recipes which user can refer to reduce weight.

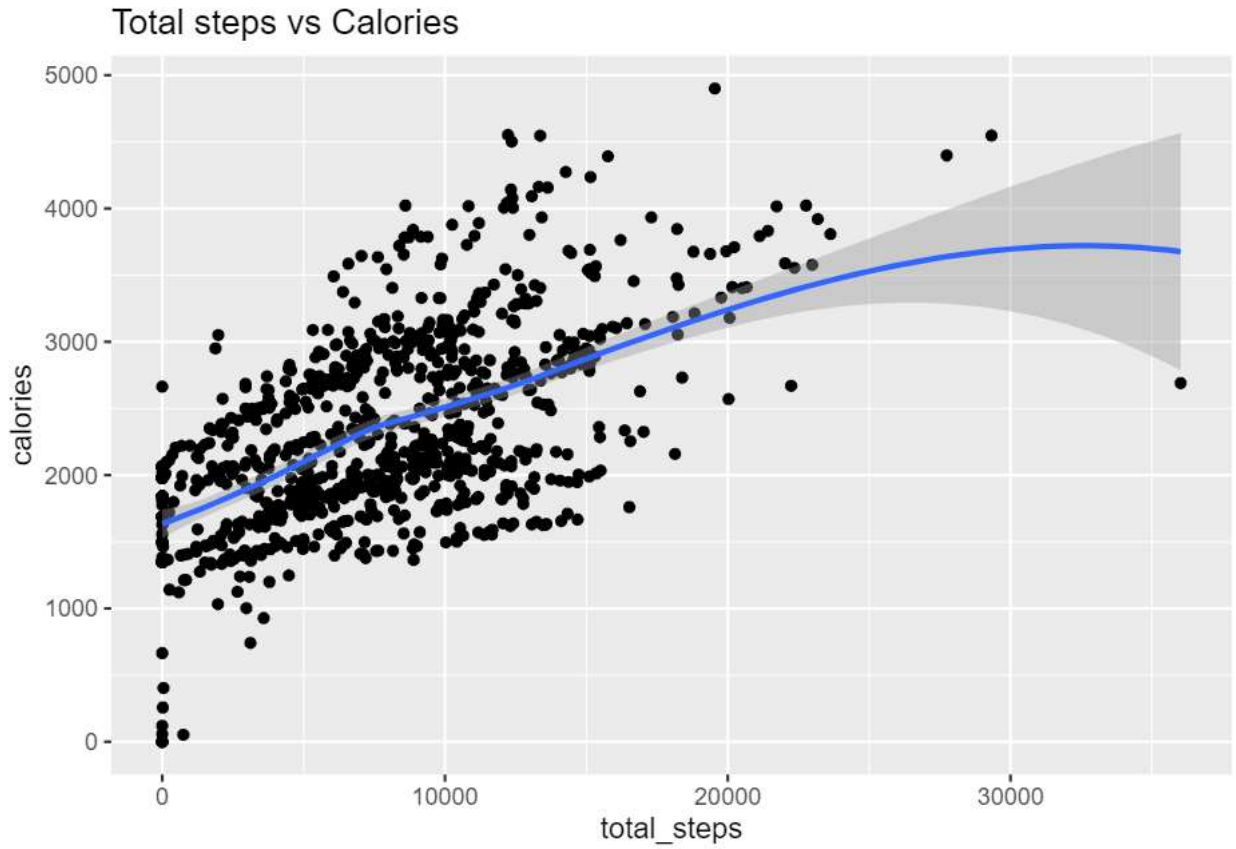
```
merged_data <- merge(activity_1, sleep_2, by = c("id", "date"))
glimpse(merged_data)
```

```
## Rows: 410
## Columns: 19
## $ id                          <dbl> 1503960366, 1503960366, 1503960366, 1503960~
## $ date                        <date> 2016-04-12, 2016-04-13, 2016-04-15, 2016-0~
## $ total_steps                 <dbl> 13162, 10735, 9762, 12669, 9705, 15506, 105~
## $ total_distance              <dbl> 8.50, 6.97, 6.28, 8.16, 6.48, 9.88, 6.68, 6~
## $ tracker_distance            <dbl> 8.50, 6.97, 6.28, 8.16, 6.48, 9.88, 6.68, 6~
## $ logged_activities_distance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ very_active_distance        <dbl> 1.88, 1.57, 2.14, 2.71, 3.19, 3.53, 1.96, 1~
## $ moderately_active_distance <dbl> 0.55, 0.69, 1.26, 0.41, 0.78, 1.32, 0.48, 0~
## $ light_active_distance       <dbl> 6.06, 4.71, 2.83, 5.04, 2.51, 5.03, 4.24, 4~
## $ sedentary_active_distance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ very_active_minutes         <dbl> 25, 21, 29, 36, 38, 50, 28, 19, 41, 39, 73,~
## $ fairly_active_minutes       <dbl> 13, 19, 34, 10, 20, 31, 12, 8, 21, 5, 14, 2~
## $ lightly_active_minutes      <dbl> 328, 217, 209, 221, 164, 264, 205, 211, 262~
## $ sedentary_minutes           <dbl> 728, 776, 726, 773, 539, 775, 818, 838, 732~
## $ calories                    <dbl> 1985, 1797, 1745, 1863, 1728, 2035, 1786, 1~
## $ time                        <Period> 12H 0M 0S, 12H 0M 0S, 12H 0M 0S, 12H 0M ~
## $ total_sleep_records         <dbl> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ total_minutes_asleep        <dbl> 327, 384, 412, 340, 700, 304, 360, 325, 361~
## $ total_time_in_bed           <dbl> 346, 407, 442, 367, 712, 320, 377, 364, 384~
```

Now we will go ahead with our visualizations

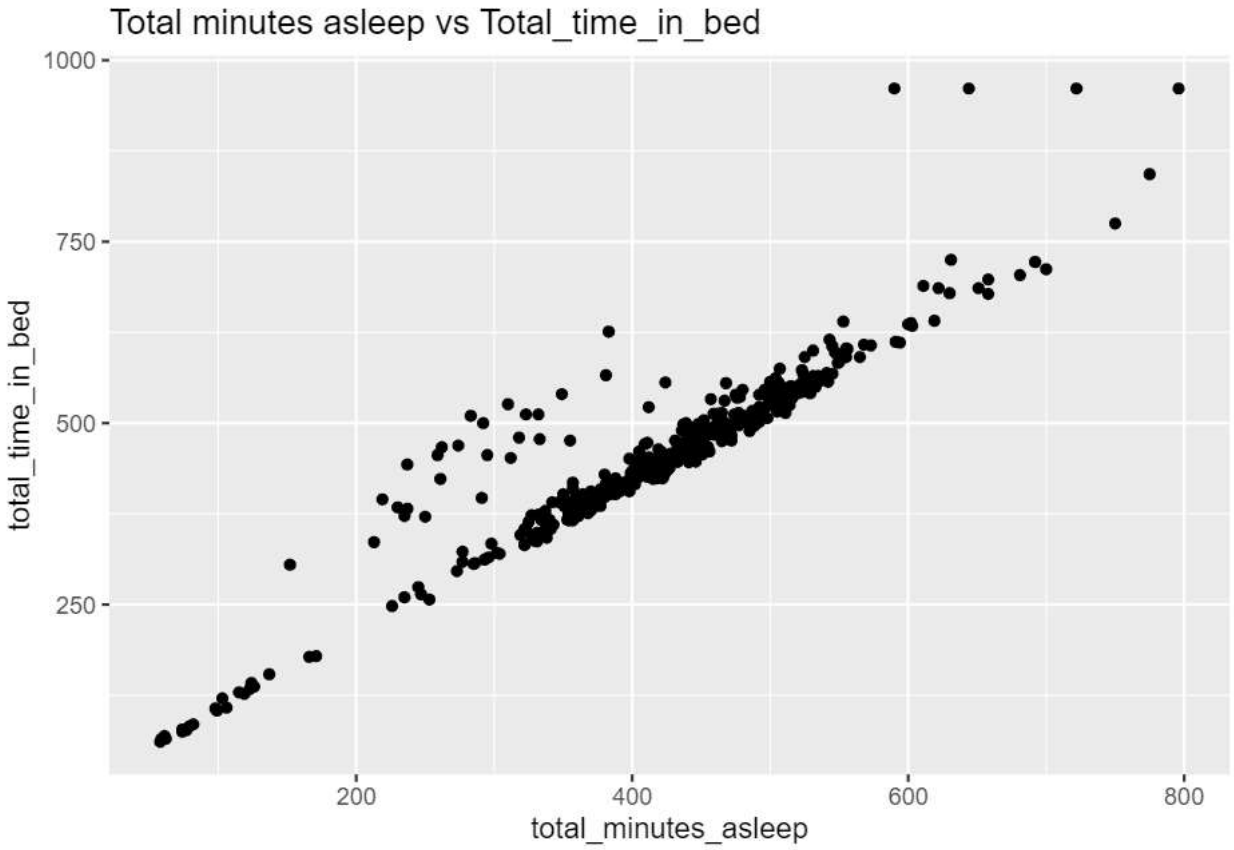
```
ggplot(data = activity_1, aes(x=total_steps, y= calories)) + geom_point() + geom_smooth() + labs(title
```

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



This shows number of steps have positive correlation with number of calories burned.

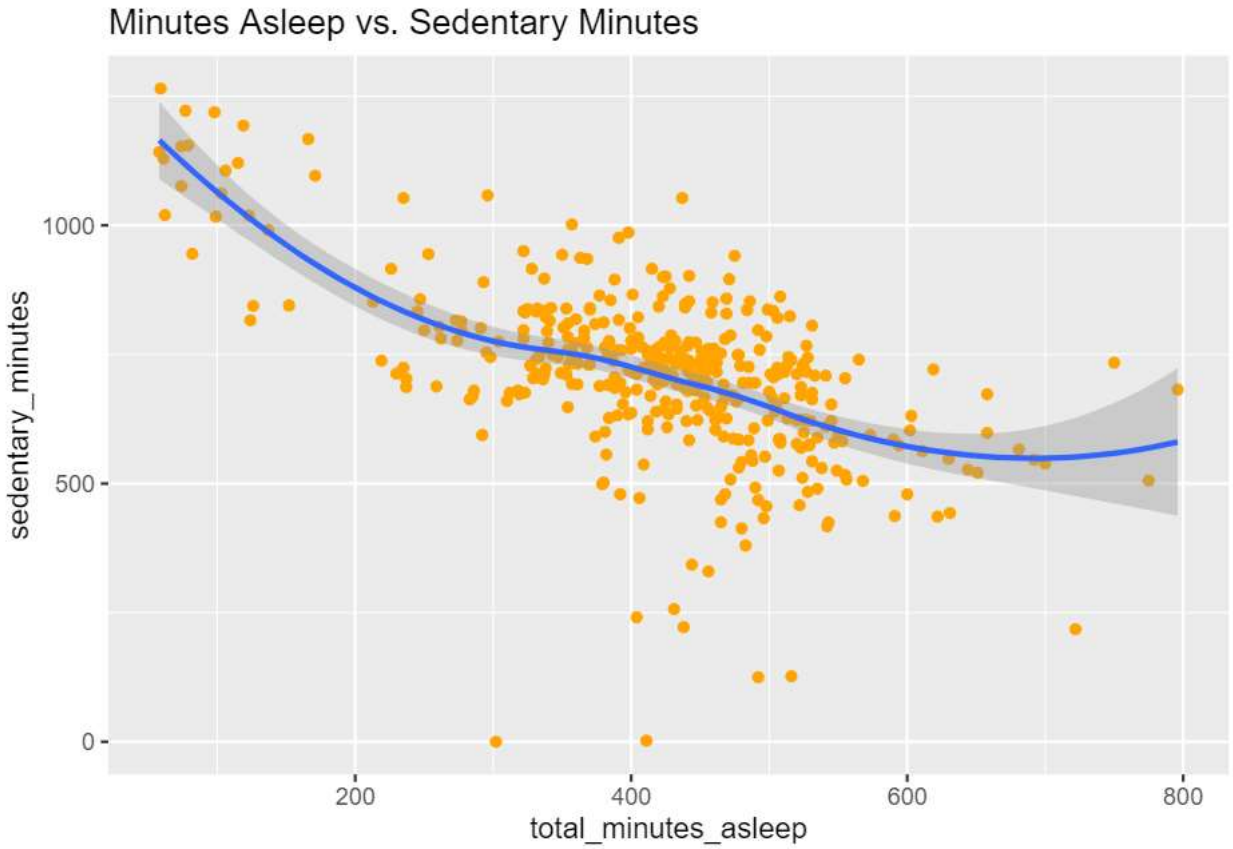
```
ggplot(data = sleep_2) + geom_point(mapping = aes(x = total_minutes_asleep, y= total_time_in_bed)) + la
```



This plot shows positive correlation between total time asleep and time in bed hence allowing user to schedule sleep routine can be useful idea.

```
ggplot(data=merged_data, aes(x=total_minutes_asleep, y=sedentary_minutes)) +
geom_point(color='orange') + geom_smooth() +
  labs(title="Minutes Asleep vs. Sedentary Minutes")
```

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



This plot shows there is negative correlation between sedentary_minutes and total minutes asleep

- This shows if bellabeat users want to improve their sleep they should reduce sedentary minutes.
- Sending idle notifications can be helpful to overcome this.

With this we have come to end of our analysis.

Key takeaways we gathered from our analysis is:

bellabeat should consider below finding based on our analysis

- Based on this we can suggest bellabeat to make strategy to motivate users for 10000 steps
- clent can start showing reminder to complete 10000 steps
- client can opt in for special customized alerts based on users previous data, if user is walking for 5 km per day generally, app should notify them if it is not done on specific day.
- Sending notifications for idle time and allowing user to snooze it or to set it for customized time can be done

Our data shows average calories burned are 2134. According to the Dietary Guidelines for Americans 2020–2025, the average adult woman burns roughly 1,600 to 2,400 calories per day so our data matches with the expectations.

- Bellabeat can use sending notifications here too to motivate users to progress on a fitness path
- Users are mostly active for 4 minutes only
- Users are fairly active for 6 minutes only
- Mostly users are lightly active
- Average user sleeps for 7 hours

- Average time in bed is also around 7.64 hours

bellabeat can use such subcategory to provide more customized user experience * bellabeat can work on creating a function which allows user to schedule sleep routine * bellabeat can also work on sending notifications when sleep routine is about to start, so that user can work on their sleeping patterns as well, because study shows that 8 hours of sleep is required for a healthy life.

- Data shows average weight of user is 62.5 KG
- Average BMI of user is 25.19, this is not good as BMI $>$ 24.9 is considered as obese
- bellabeat can work on suggesting low calorie diets or recipes which user can refer to reduce weight.
- Data shows if bellabeat users want to improve their sleep they should reduce sedentary minutes.
- Sending idle notifications can be helpful to overcome this

Thank you for your time, please share your valuable feedback!