Capstone Project for the Google Data Analytics Professional Certificate

Bellabeat is a high-tech manufacturer of health-focused products for women. As a junior data analyst working with marketing analyst team at Bellabeat, a high-tech manufacturer of health-focused products for women. have been asked to focus on one of Bellabeat's products and analyze smart device data to gain insight into how consumers are using their smart devices. I have performed analysis on data to give recommendations.

1. Ask

The business task is to analyze smart device usage data to gain insights, identify trends and understand how users use these products. The insights will be used to make data-driven decisions by applying them to one of Bellabeat's products and help guide marketing strategy.

The key stakeholders are as follows:

- Urška Sršen: Bellabeat's co-founder and Chief Creative Officer
- Sando Mur: Mathematician and Bellabeat's cofounder; a key member of Bellabeat's executive team
- Bellabeat's marketing analytics team: A team of data analysts responsible for collecting, analyzing, and reporting data that helps guide Bellabeat's marketing strategy.

Business objectives are as follows:

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's marketing strategy?

2. Prepare

The data set is Fitbit Fitness Tracker Data taken from Kaggle which contains personal fitness trackers from thirty Fitbit users. It contains 18 CSV files. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It includes information about daily activity, steps, and heart rate that can be used to explore users' habits. Generated by respondents from a survey via Amazon Mechanical Turk between 12 March 2016 to 12 May 2016.

The dataset is organized in long and wide formats. The dataset is bad quality as we can check for the Dataset to be ROCCC as follows.

- Reliable: Low, as there are only 30 respondents
- Original: Low as the third-party provider (Amazon Mechanical Turk)
- Comprehensive: Medium, as matches Bellabeat product data
- Current: Low, data is 5 years old
- Cited: Low, Data collected from third party

3. Process, Analyze, Share, Act

Excel will be used for the data cleaning process and for removing errors from it. R programming language will be used for analysis and visualization.

Following cleaning and manipulation of data are handled. Null data, misspelled words, mistyped numbers, extra spaces and characters, duplication, mismatched data types, inconsistent strings, and date formats, misleading column names, business logic, and truncated/missing data.

The data has been uploaded to R and analyzed.

import library and datasets available in folder

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.0     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.1     ✓ tibble    3.1.8
## ✓ lubridate 1.9.2     ✓ tidyr     1.3.0
## ✓ purrr     1.0.1
## ── Conflicts ────────────────────────────────────────── tidyverse_confl
icts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to for
ce all conflicts to become errors
```

```
# install.packages("dplyr", repos = "http://cran.us.r-project.org")
library(dplyr)
# install.packages("janitor", repos = "http://cran.us.r-project.org")
library(janitor)
```

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

```
library(ggplot2)
library(lubridate)
install.packages("openxlsx", dependencies=TRUE, repos = "http://cran.us.r-
project.org")
```

```
## Installing package into 'C:/Users/tashf/AppData/Local/R/win-library/4.2
'
## (as 'lib' is unspecified)

## package 'openxlsx' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'openxlsx'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\tashf\AppData\Local\R\win-library\4.2\00LOCK\openxlsx\libs\x64
\openxlsx.dll
## to
## C:\Users\tashf\AppData\Local\R\win-library\4.2\openxlsx\libs\x64\openxl
sx.dll:
## Permission denied

## Warning: restored 'openxlsx'

##
## The downloaded binary packages are in
##  C:\Users\tashf\AppData\Local\Temp\RtmpOg0lKY\downloaded_packages
```

Importing data

```
daily_activity <- read.csv('dailyActivity_merged.csv')

daily_sleep <- read.csv('sleepDay_merged.csv')

weight_log <- read.csv('weightLogInfo_merged.csv')

daily_calories <- read.csv('dailyCalories_merged.csv')

daily_intensities <- read.csv('dailyIntensities_merged.csv')

daily_steps <- read.csv('dailySteps_merged.csv')

hourly_steps  <- read.csv('hourlySteps_merged.csv')

hourly_calories <- read.csv('hourlyCalories_merged.csv')
```

Preview Data

```
head(daily_activity)
```

```
##           Id      Date  Time WeightKg WeightPounds Fat   BMI IsManualRe
port
## 1 1503960366  5/2/2016 23:59     52.6     115.9631  22 22.65
TRUE
## 2 1503960366  5/3/2016 23:59     52.6     115.9631  NA 22.65
TRUE
## 3 1927972279 4/13/2016  1:08    133.5     294.3171  NA 47.54          F
ALSE
## 4 2873212765 4/21/2016 23:59     56.7     125.0021  NA 21.45
TRUE
## 5 2873212765 5/12/2016 23:59     57.3     126.3249  NA 21.69
TRUE
## 6 4319703577 4/17/2016 23:59     72.4     159.6147  25 27.45
TRUE
##      LogId
## 1 1.46e+12
## 2 1.46e+12
## 3 1.46e+12
## 4 1.46e+12
```

```
## 5 1.46e+12
## 6 1.46e+12
```

```
head(daily_sleep)
```

```
##           Id  SleepDay TotalSleepRecords TotalMinutesAsleep TotalTimeIn
Bed
## 1 1503960366 4/12/2016                 1                327
346
## 2 1503960366 4/13/2016                 2                384
407
## 3 1503960366 4/15/2016                 1                412
442
## 4 1503960366 4/16/2016                 2                340
367
## 5 1503960366 4/17/2016                 1                700
712
## 6 1503960366 4/19/2016                 1                304
320
```

```
head(daily_calories)
```

```
##           Id ActivityDay Calories
## 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
```

```
head(weight_log)
```

```
##           Id      Date WeightKg WeightPounds Fat   BMI IsManualReport
## 1 1503960366  5/2/2016     52.6     115.9631  22 22.65           TRUE
## 2 1503960366  5/3/2016     52.6     115.9631  NA 22.65           TRUE
## 3 1927972279 4/13/2016    133.5     294.3171  NA 47.54          FALSE
## 4 2873212765 4/21/2016     56.7     125.0021  NA 21.45           TRUE
## 5 2873212765 5/12/2016     57.3     126.3249  NA 21.69           TRUE
## 6 4319703577 4/17/2016     72.4     159.6147  25 27.45           TRUE
##         LogId  X
## 1 1.46223e+12 NA
## 2 1.46232e+12 NA
## 3 1.46051e+12 NA
## 4 1.46128e+12 NA
## 5 1.46310e+12 NA
## 6 1.46094e+12 NA
```

```
head(hourly_steps)
```

```
##           Id          ActivityHour StepTotal
## 1 1503960366 4/12/2016 12:00:00 AM       373
## 2 1503960366  4/12/2016 1:00:00 AM       160
## 3 1503960366  4/12/2016 2:00:00 AM       151
## 4 1503960366  4/12/2016 3:00:00 AM         0
```

```
## 5 1503960366  4/12/2016 4:00:00 AM         0
## 6 1503960366  4/12/2016 5:00:00 AM         0
```

```
head(hourly_calories)
```

```
##           Id          ActivityHour Calories
## 1 1503960366 4/12/2016 12:00:00 AM       81
## 2 1503960366  4/12/2016 1:00:00 AM       61
## 3 1503960366  4/12/2016 2:00:00 AM       59
## 4 1503960366  4/12/2016 3:00:00 AM       47
## 5 1503960366  4/12/2016 4:00:00 AM       48
## 6 1503960366  4/12/2016 5:00:00 AM       48
```

```
str(daily_activity)
```

```
## 'data.frame':    67 obs. of  9 variables:
##  $ Id            : num  1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ..
.
##  $ Date          : chr  "5/2/2016" "5/3/2016" "4/13/2016" "4/21/2016" .
..
##  $ Time          : chr  "23:59" "23:59" "1:08" "23:59" ...
##  $ WeightKg      : num  52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num  116 116 294 125 126 ...
##  $ Fat           : int  22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num  22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: logi  TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num  1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ..
.
```

```
str(daily_sleep)
```

```
## 'data.frame':    413 obs. of  5 variables:
##  $ Id                : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ SleepDay          : chr  "4/12/2016" "4/13/2016" "4/15/2016" "4/16/2
016" ...
##  $ TotalSleepRecords : int  1 2 1 2 1 1 1 1 1 1 ...
##  $ TotalMinutesAsleep: int  327 384 412 340 700 304 360 325 361 430 ...
##  $ TotalTimeInBed    : int  346 407 442 367 712 320 377 364 384 449 ...
```

```
str(hourly_calories)
```

```
## 'data.frame':    22099 obs. of  3 variables:
##  $ Id          : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityHour: chr  "4/12/2016 12:00:00 AM" "4/12/2016 1:00:00 AM" "4
/12/2016 2:00:00 AM" "4/12/2016 3:00:00 AM" ...
##  $ Calories    : int  81 61 59 47 48 48 48 47 68 141 ...
```

```
str(weight_log)
```

```
## 'data.frame':    67 obs. of  9 variables:
##  $ Id            : num  1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ..
.
##  $ Date          : chr  "5/2/2016" "5/3/2016" "4/13/2016" "4/21/2016" .
..
##  $ WeightKg      : num  52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num  116 116 294 125 126 ...
```

```
##  $ Fat           : int  22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num  22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: logi  TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num  1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ..
.
##  $ X             : logi  NA NA NA NA NA NA ...
```

```
str(hourly_steps)
```

```
## 'data.frame':    22099 obs. of  3 variables:
##  $ Id          : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityHour: chr  "4/12/2016 12:00:00 AM" "4/12/2016 1:00:00 AM" "4
/12/2016 2:00:00 AM" "4/12/2016 3:00:00 AM" ...
##  $ StepTotal   : int  373 160 151 0 0 0 0 0 250 1864 ...
```

```
str(hourly_calories)
```

```
## 'data.frame':    22099 obs. of  3 variables:
##  $ Id          : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityHour: chr  "4/12/2016 12:00:00 AM" "4/12/2016 1:00:00 AM" "4
/12/2016 2:00:00 AM" "4/12/2016 3:00:00 AM" ...
##  $ Calories    : int  81 61 59 47 48 48 48 47 68 141 ...
```

To confirm a number of participants in all data sets I will count unique IDs.

```
n_distinct(weight_log$Id)
```

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

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

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

```
n_distinct(daily_sleep$Id)
```

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

```
n_distinct(daily_calories$Id)
```

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

```
n_distinct(hourly_steps$Id)
```

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

```
n_distinct(hourly_calories$Id)
```

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

checking a total number of duplicates in each data frame.

```
sum(duplicated(weight_log))
```

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

```
sum(duplicated(daily_activity))
```

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

```
sum(duplicated(daily_sleep))
```

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

```
sum(duplicated(daily_calories))
```

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

```
sum(duplicated(hourly_steps))
```

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

```
sum(duplicated(hourly_calories))
```

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

Only sleep data contains duplicates. They need to be cleaned.

```
daily_sleep <- daily_sleep %>%
  distinct()
sum(duplicated(daily_sleep))
```

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

Standardizing column names

```
# clean_names(daily_activity)
# clean_names(daily_calories)
# clean_names(hourly_calories)
# clean_names(daily_sleep)
# clean_names(hourly_steps)
```

Format Date and Time

Change variable name

```
daily_activity <- daily_activity %>%
    rename(ActivityDay = Date)

daily_activity <- daily_activity %>%
  mutate(ActivityDay = as_date(ActivityDay, format = "%m/%d/%Y"))

daily_activity <- daily_activity %>%
           mutate(year = lubridate::year(ActivityDay),
                month = lubridate::month(ActivityDay),
                day = lubridate::day(ActivityDay))

daily_calories <- daily_calories %>%
  mutate(ActivityDay = as.Date(ActivityDay, format = "%m/%d/%Y")) %>%
  mutate(year = lubridate::year(ActivityDay),
                month = lubridate::month(ActivityDay),
                day = lubridate::day(ActivityDay))

daily_intensities <- daily_intensities %>%
  mutate(ActivityDay = as.Date(ActivityDay, format = "%m/%d/%Y")) %>%
  mutate(year = lubridate::year(ActivityDay),
                month = lubridate::month(ActivityDay),
                day = lubridate::day(ActivityDay))
```

```
daily_sleep <- daily_sleep %>%
  rename(ActivityDay = SleepDay) %>%
  mutate(ActivityDay = as.Date(ActivityDay, format = "%m/%d/%Y")) %>%
  mutate(year = lubridate::year(ActivityDay),
                month = lubridate::month(ActivityDay),
                day = lubridate::day(ActivityDay))

daily_steps <- daily_steps %>%
  mutate(ActivityDay = as.Date(ActivityDay, format = "%m/%d/%Y")) %>%
  mutate(year = lubridate::year(ActivityDay),
                month = lubridate::month(ActivityDay),
                day = lubridate::day(ActivityDay))

weight_log <- weight_log %>%
  rename(ActivityDay = Date) %>%
  mutate(ActivityDay = as.Date(ActivityDay, format = "%m/%d/%Y")) %>%
           mutate(year = lubridate::year(ActivityDay),
                  month = lubridate::month(ActivityDay),
                  day = lubridate::day(ActivityDay))
```

Format date/time data where a time stamp is in 12 hours AM/PM format

```
daily_activity <- daily_activity %>%
  mutate(Time = format(strptime(Time, "%H:%M"), "%I:%M"))
```

Analyze join using id and date columns

```
merge_1 <- full_join(daily_activity, daily_sleep, by = c("Id", "year", "mo
nth", "day" ))
merge_2 <- full_join(daily_steps, daily_calories, by = c("Id", "year", "mo
nth", "day" ))
merge_3 <- full_join(daily_intensities, weight_log, by = c("Id", "year", "
month", "day" ))
merge_4 <- full_join(merge_1, merge_2, by = c("Id", "year", "month", "day"
))

merge_5 <- full_join(merge_3,merge_4, by = c("Id", "year", "month", "day"
))
```

summarize the data.

```
summary(merge_5)
```

```
##        Id             ActivityDay.x        SedentaryMinutes LightlyActiv
eMinutes
##  Min.   :1.504e+09   Min.   :2016-04-12   Min.   :   0.0   Min.   :  0.
0
##  1st Qu.:2.320e+09   1st Qu.:2016-04-19   1st Qu.: 729.8   1st Qu.:127.
0
##  Median :4.445e+09   Median :2016-04-26   Median :1057.5   Median :199.
0
##  Mean   :4.855e+09   Mean   :2016-04-26   Mean   : 991.2   Mean   :192.
8
##  3rd Qu.:6.962e+09   3rd Qu.:2016-05-04   3rd Qu.:1229.5   3rd Qu.:264.
0
##  Max.   :8.878e+09   Max.   :2016-05-12   Max.   :1440.0   Max.   :518.
```

```
##
##  FairlyActiveMinutes VeryActiveMinutes SedentaryActiveDistance
##  Min.   :  0.00      Min.   :  0.00    Min.   :0.000000
##  1st Qu.:  0.00      1st Qu.:  0.00    1st Qu.:0.000000
##  Median :  6.00      Median :  4.00    Median :0.000000
##  Mean   : 13.56      Mean   : 21.16    Mean   :0.001606
##  3rd Qu.: 19.00      3rd Qu.: 32.00    3rd Qu.:0.000000
##  Max.   :143.00      Max.   :210.00    Max.   :0.110000
##
##  LightActiveDistance ModeratelyActiveDistance VeryActiveDistance      y
ear
##  Min.   : 0.000      Min.   :0.0000           Min.   : 0.000     Min.
:2016
##  1st Qu.: 1.945      1st Qu.:0.0000           1st Qu.: 0.000     1st Qu
.:2016
##  Median : 3.365      Median :0.2400           Median : 0.210     Median
:2016
##  Mean   : 3.341      Mean   :0.5675           Mean   : 1.503     Mean
:2016
##  3rd Qu.: 4.782      3rd Qu.:0.8000           3rd Qu.: 2.053     3rd Qu
.:2016
##  Max.   :10.710      Max.   :6.4800           Max.   :21.920     Max.
:2016
##
##      month           day        ActivityDay.y          WeightKg.x
##  Min.   :4.00   Min.   : 1.00   Min.   :2016-04-12   Min.   : 52.60
##  1st Qu.:4.00   1st Qu.: 9.00   1st Qu.:2016-04-19   1st Qu.: 61.40
##  Median :4.00   Median :16.00   Median :2016-04-27   Median : 62.50
##  Mean   :4.35   Mean   :15.79   Mean   :2016-04-26   Mean   : 72.04
##  3rd Qu.:5.00   3rd Qu.:23.00   3rd Qu.:2016-05-04   3rd Qu.: 85.05
##  Max.   :5.00   Max.   :30.00   Max.   :2016-05-12   Max.   :133.50
##                                 NA's   :873          NA's   :873
##  WeightPounds.x      Fat.x           BMI.x       IsManualReport.x
##  Min.   :116.0   Min.   :22.00   Min.   :21.45   Mode :logical
##  1st Qu.:135.4   1st Qu.:22.75   1st Qu.:23.96   FALSE:26
##  Median :137.8   Median :23.50   Median :24.39   TRUE :41
##  Mean   :158.8   Mean   :23.50   Mean   :25.19   NA's :873
##  3rd Qu.:187.5   3rd Qu.:24.25   3rd Qu.:25.56
##  Max.   :294.3   Max.   :25.00   Max.   :47.54
##  NA's   :873     NA's   :938     NA's   :873
##     LogId.x             X           ActivityDay.x.x          Time
##  Min.   :1.460e+12   Mode:logical   Min.   :2016-04-12   Length:940
##  1st Qu.:1.461e+12   NA's:940       1st Qu.:2016-04-19   Class :charact
er
##  Median :1.462e+12                  Median :2016-04-27   Mode  :charact
er
##  Mean   :1.462e+12                  Mean   :2016-04-26
##  3rd Qu.:1.462e+12                  3rd Qu.:2016-05-04
##  Max.   :1.463e+12                  Max.   :2016-05-12
##  NA's   :873                        NA's   :873
##    WeightKg.y     WeightPounds.y      Fat.y           BMI.y
##  Min.   : 52.60   Min.   :116.0   Min.   :22.00   Min.   :21.45
##  1st Qu.: 61.40   1st Qu.:135.4   1st Qu.:22.75   1st Qu.:23.96
```

```
##  Median : 62.50   Median :137.8   Median :23.50   Median :24.39
##  Mean   : 72.04   Mean   :158.8   Mean   :23.50   Mean   :25.19
##  3rd Qu.: 85.05   3rd Qu.:187.5   3rd Qu.:24.25   3rd Qu.:25.56
##  Max.   :133.50   Max.   :294.3   Max.   :25.00   Max.   :47.54
##  NA's   :873      NA's   :873     NA's   :938     NA's   :873
##  IsManualReport.y    LogId.y         ActivityDay.y.x      TotalSleepRec
ords
##  Mode :logical    Min.   :1.46e+12   Min.   :2016-04-12   Min.   :1.000
##  FALSE:26         1st Qu.:1.46e+12   1st Qu.:2016-04-19   1st Qu.:1.000
##  TRUE :41         Median :1.46e+12   Median :2016-04-27   Median :1.000
##  NA's :873        Mean   :1.46e+12   Mean   :2016-04-26   Mean   :1.119
##                   3rd Qu.:1.46e+12   3rd Qu.:2016-05-04   3rd Qu.:1.000
##                   Max.   :1.46e+12   Max.   :2016-05-12   Max.   :3.000
##                   NA's   :873        NA's   :530          NA's   :530
##  TotalMinutesAsleep TotalTimeInBed  ActivityDay.x.y        StepTotal
##  Min.   : 58.0      Min.   : 61.0   Min.   :2016-04-12   Min.   :    0
##  1st Qu.:361.0      1st Qu.:403.8   1st Qu.:2016-04-19   1st Qu.: 3790
##  Median :432.5      Median :463.0   Median :2016-04-26   Median : 7406
##  Mean   :419.2      Mean   :458.5   Mean   :2016-04-26   Mean   : 7638
##  3rd Qu.:490.0      3rd Qu.:526.0   3rd Qu.:2016-05-04   3rd Qu.:10727
##  Max.   :796.0      Max.   :961.0   Max.   :2016-05-12   Max.   :36019
##  NA's   :530        NA's   :530
##  ActivityDay.y.y         Calories
##  Min.   :2016-04-12   Min.   :   0
##  1st Qu.:2016-04-19   1st Qu.:1828
##  Median :2016-04-26   Median :2134
##  Mean   :2016-04-26   Mean   :2304
##  3rd Qu.:2016-05-04   3rd Qu.:2793
##  Max.   :2016-05-12   Max.   :4900
##
```

check data for unique id and duplicate

```
 n_distinct(merge_5$Id)

## [1] 33

nrow(merge_5)

## [1] 940
```

should be 30 as survey was of 30 people

```
 # clean_names(merge_5)

daily_sleep %>%
  select(TotalSleepRecords,
  TotalMinutesAsleep,
  TotalTimeInBed) %>%
  summary()

##  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##  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

colnames(merge_5)

##  [1] "Id"                       "ActivityDay.x"
##  [3] "SedentaryMinutes"         "LightlyActiveMinutes"
##  [5] "FairlyActiveMinutes"      "VeryActiveMinutes"
##  [7] "SedentaryActiveDistance"  "LightActiveDistance"
##  [9] "ModeratelyActiveDistance" "VeryActiveDistance"
## [11] "year"                     "month"
## [13] "day"                      "ActivityDay.y"
## [15] "WeightKg.x"               "WeightPounds.x"
## [17] "Fat.x"                    "BMI.x"
## [19] "IsManualReport.x"         "LogId.x"
## [21] "X"                        "ActivityDay.x.x"
## [23] "Time"                     "WeightKg.y"
## [25] "WeightPounds.y"           "Fat.y"
## [27] "BMI.y"                    "IsManualReport.y"
## [29] "LogId.y"                  "ActivityDay.y.x"
## [31] "TotalSleepRecords"        "TotalMinutesAsleep"
## [33] "TotalTimeInBed"           "ActivityDay.x.y"
## [35] "StepTotal"                "ActivityDay.y.y"
## [37] "Calories"

install.packages("rio", repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/tashf/AppData/Local/R/win-library/4.2
'
## (as 'lib' is unspecified)

## package 'rio' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\tashf\AppData\Local\Temp\RtmpOg0lKY\downloaded_packages

library(openxlsx)
library(rio)

export(merge_5, "merge_5_to_clean.xlsx")

merge_6 <- subset(merge_5, select = -c(WeightKg.y
, ActivityDay.y
, Fat.y
, LogId.y
, ActivityDay.y.x
, ActivityDay.x.y
, ActivityDay.y.y
,ActivityDay.x.x, BMI.y,WeightPounds.y,IsManualReport.y))

colnames(merge_6)

##  [1] "Id"                       "ActivityDay.x"
##  [3] "SedentaryMinutes"         "LightlyActiveMinutes"
##  [5] "FairlyActiveMinutes"      "VeryActiveMinutes"
```

```
##  [7] "SedentaryActiveDistance"  "LightActiveDistance"
##  [9] "ModeratelyActiveDistance" "VeryActiveDistance"
## [11] "year"                     "month"
## [13] "day"                      "WeightKg.x"
## [15] "WeightPounds.x"           "Fat.x"
## [17] "BMI.x"                    "IsManualReport.x"
## [19] "LogId.x"                  "X"
## [21] "Time"                     "TotalSleepRecords"
## [23] "TotalMinutesAsleep"       "TotalTimeInBed"
## [25] "StepTotal"                "Calories"
```

```
setwd("Created DataSets")
export(merge_6, "MERGE_6_CLEANED_DATA.XLSX")
```

```
str(merge_6)
```

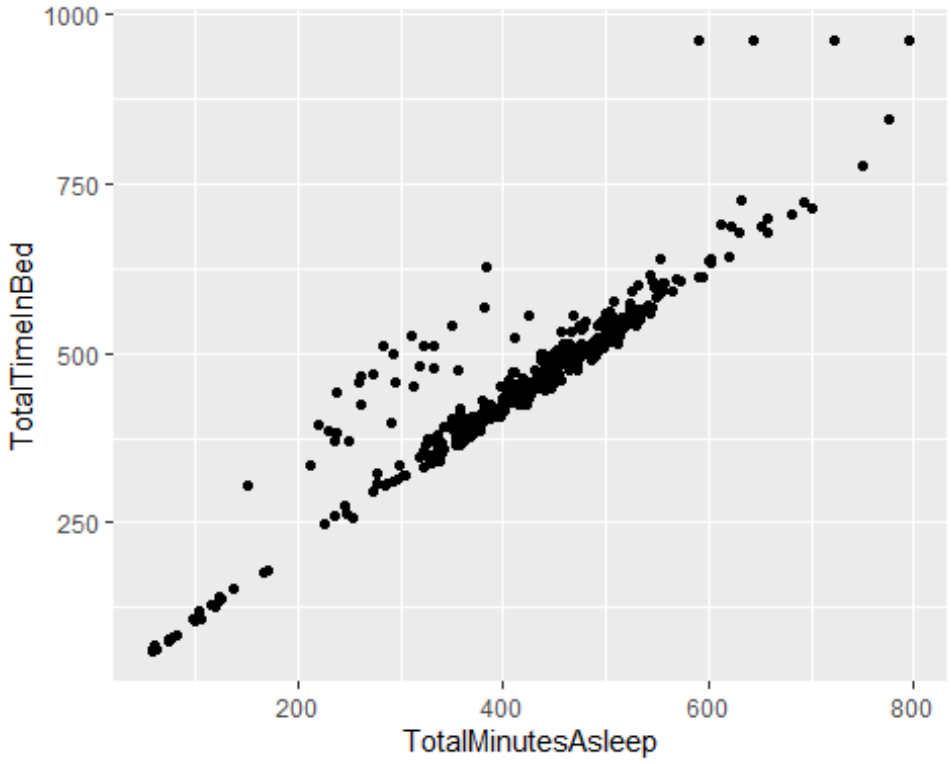
```
## 'data.frame':    940 obs. of  26 variables:
##  $ Id                       : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+
09 ...
##  $ ActivityDay.x            : Date, format: "2016-04-12" "2016-04-13" ..
.
##  $ SedentaryMinutes         : int  728 776 1218 726 773 539 1149 775 818
838 ...
##  $ LightlyActiveMinutes     : int  328 217 181 209 221 164 233 264 205 2
11 ...
##  $ FairlyActiveMinutes      : int  13 19 11 34 10 20 16 31 12 8 ...
##  $ VeryActiveMinutes        : int  25 21 30 29 36 38 42 50 28 19 ...
##  $ SedentaryActiveDistance : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ LightActiveDistance      : num  6.06 4.71 3.91 2.83 5.04 ...
##  $ ModeratelyActiveDistance: num  0.55 0.69 0.4 1.26 0.41 ...
##  $ VeryActiveDistance       : num  1.88 1.57 2.44 2.14 2.71 ...
##  $ year                     : num  2016 2016 2016 2016 2016 ...
##  $ month                    : num  4 4 4 4 4 4 4 4 4 4 ...
##  $ day                      : int  12 13 14 15 16 17 18 19 20 21 ...
##  $ WeightKg.x               : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ WeightPounds.x           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ Fat.x                    : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ BMI.x                    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ IsManualReport.x         : logi  NA NA NA NA NA NA ...
##  $ LogId.x                  : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ X                        : logi  NA NA NA NA NA NA ...
##  $ Time                     : chr  NA NA NA NA ...
##  $ TotalSleepRecords        : int  1 2 NA 1 2 1 NA 1 1 1 ...
##  $ TotalMinutesAsleep       : int  327 384 NA 412 340 700 NA 304 360 325
...
##  $ TotalTimeInBed           : int  346 407 NA 442 367 712 NA 320 377 364
...
##  $ StepTotal                : int  13162 10735 10460 9762 12669 9705 130
19 15506 10544 9819 ...
##  $ Calories                 : int  1985 1797 1776 1745 1863 1728 1921 20
35 1786 1775 ...
```

Average Table

```
 average_values1 <- data.frame(avg_sedentary_mins =mean(merge_6$SedentaryM
inutes),

avg_light_active_mins = mean(merge_6$LightlyActiveMinutes),

avg_fairly_active_mins = mean(merge_6$FairlyActiveMinutes),

avg_very_active_mins = mean(merge_6$VeryActiveMinutes),

avg_sedentary_active_distance = mean(merge_6$SedentaryActiveDistance),

avg_light_active_distance = 3.34,

avg_moderately_active_distance = mean(merge_6$ModeratelyActiveDistance),

avg_very_active_distance = mean(merge_6$VeryActiveDistance),

avg_minutes_sleep = mean(merge_6$TotalMinutesAsleep, na.rm = TRUE),

avg_calories = mean(merge_6$Calories),

avg_steps = mean(merge_6$StepTotal))

head(average_values1)

##   avg_sedentary_mins avg_light_active_mins avg_fairly_active_mins
## 1           991.2106              192.8128               13.56489
##   avg_very_active_mins avg_sedentary_active_distance avg_light_active_d
istance
## 1             21.16489                   0.001606383
3.34
##   avg_moderately_active_distance avg_very_active_distance avg_minutes_s
leep
## 1                      0.5675426                 1.502681          419.
1732
##   avg_calories avg_steps
## 1      2303.61  7637.911

export(average_values1, "Partial Average Values.xlsx")
```

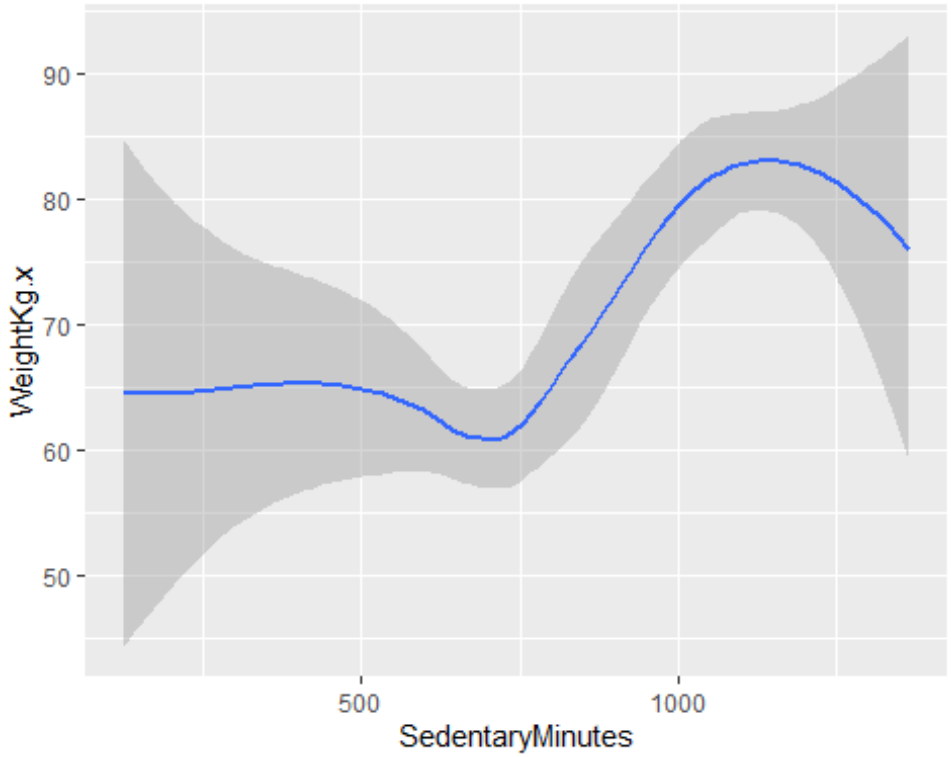
Plot Graphs

```
ggplot(data=merge_6) +geom_point(mapping=aes(x=TotalMinutesAsleep,y=TotalT
imeInBed))

## Warning: Removed 530 rows containing missing values (`geom_point()`).
```

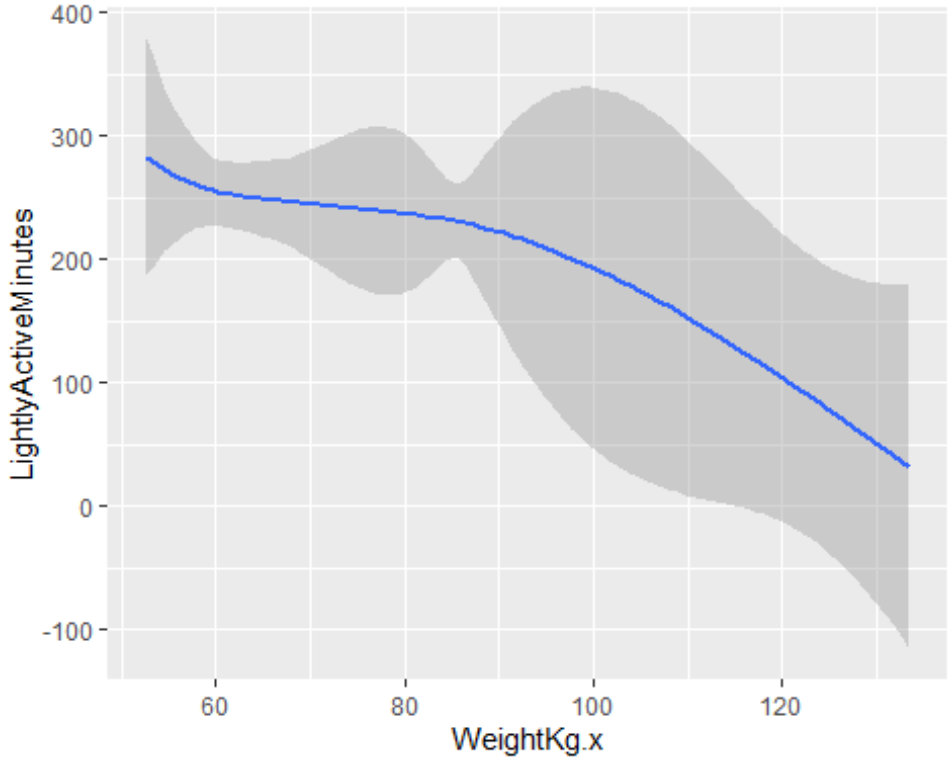


Linear trend found which is expected

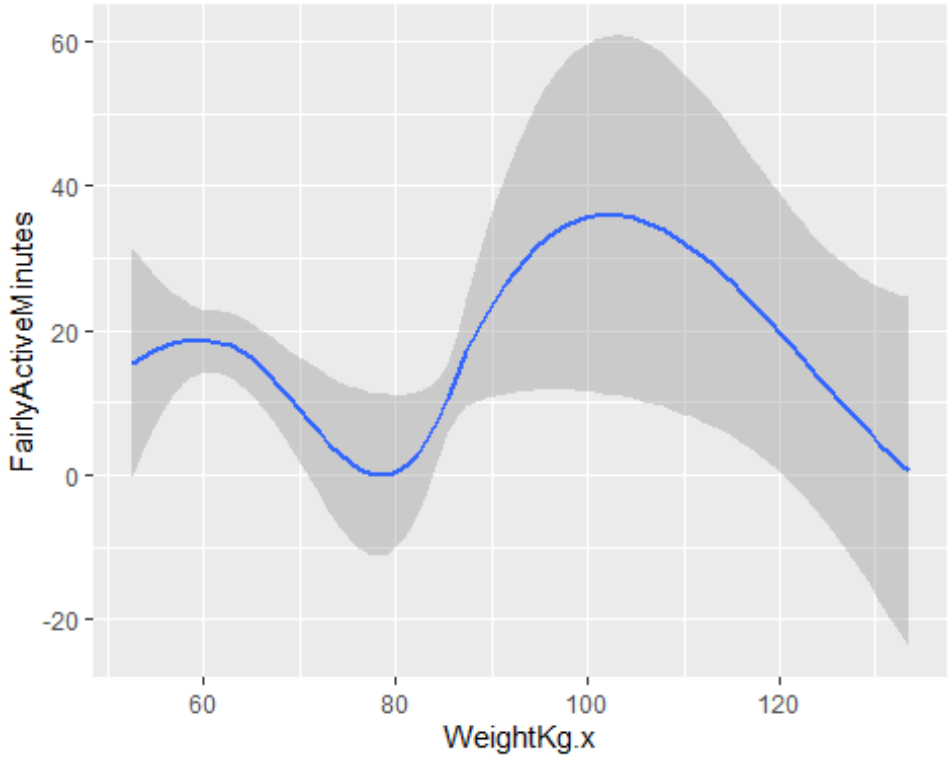
```
ggplot(data=merge_6) +geom_smooth(mapping=aes(y=WeightKg.x, x = SedentaryMinutes))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`).
```



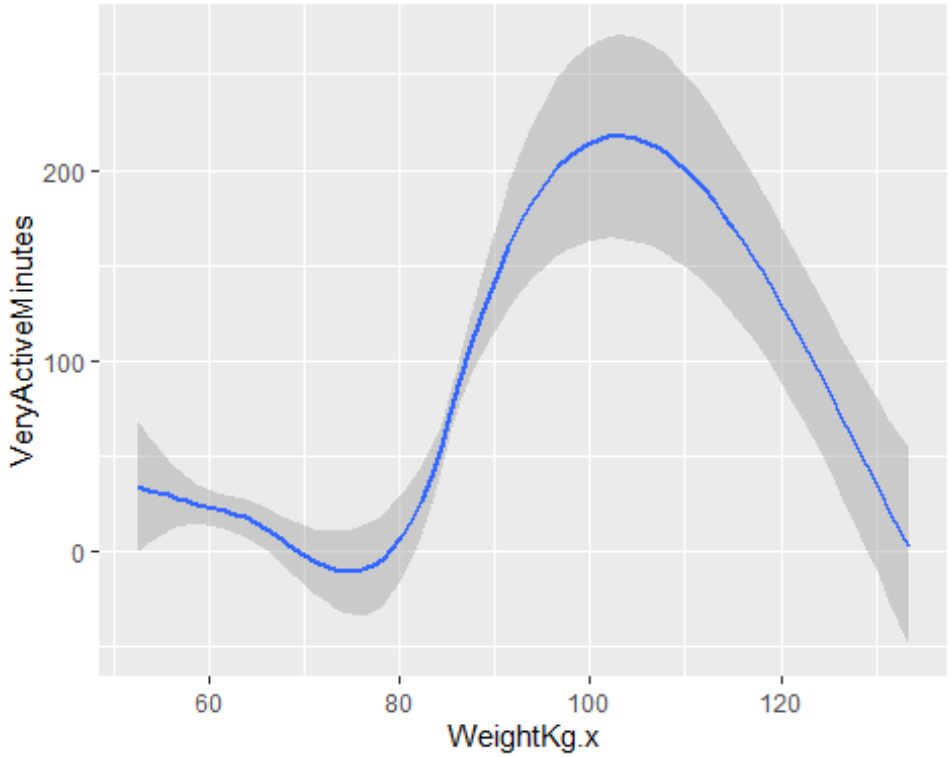

```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = LightlyActiveMinutes))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`).
```



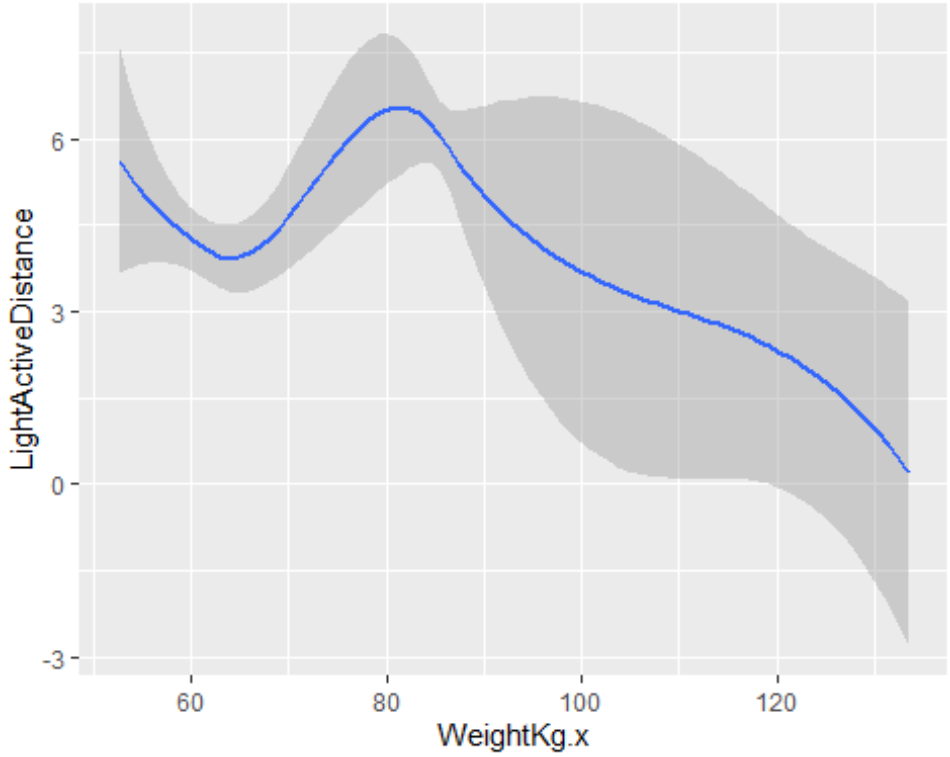
```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = FairlyActi
veMinutes))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```



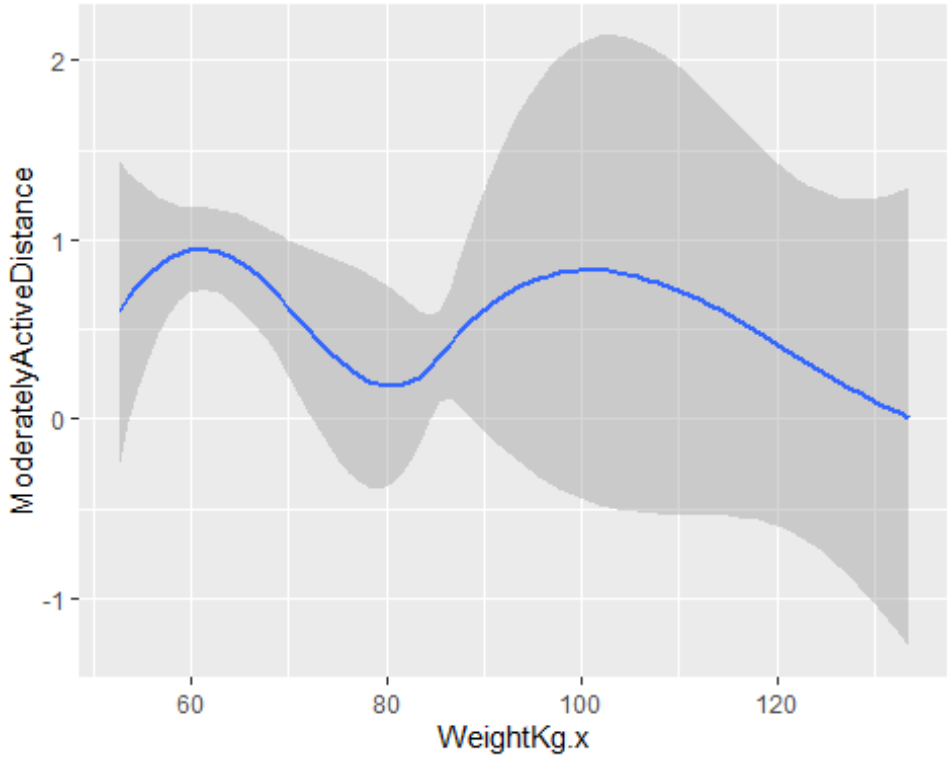

```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = VeryActive
Minutes))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```



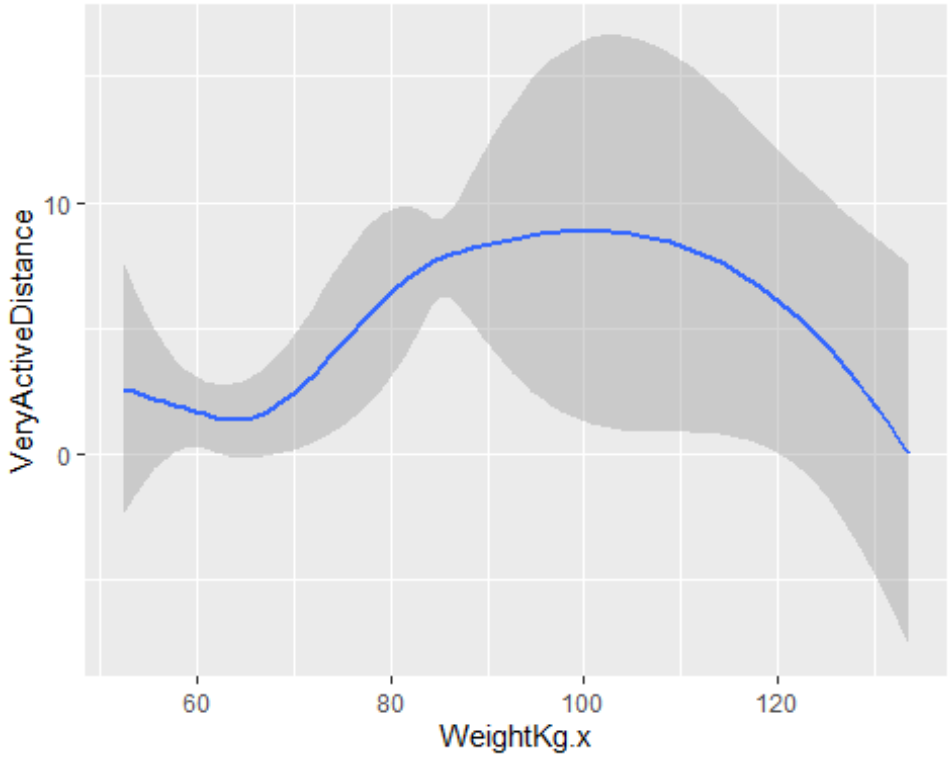

```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = LightActiv
eDistance))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```



```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = Moderately
ActiveDistance))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```

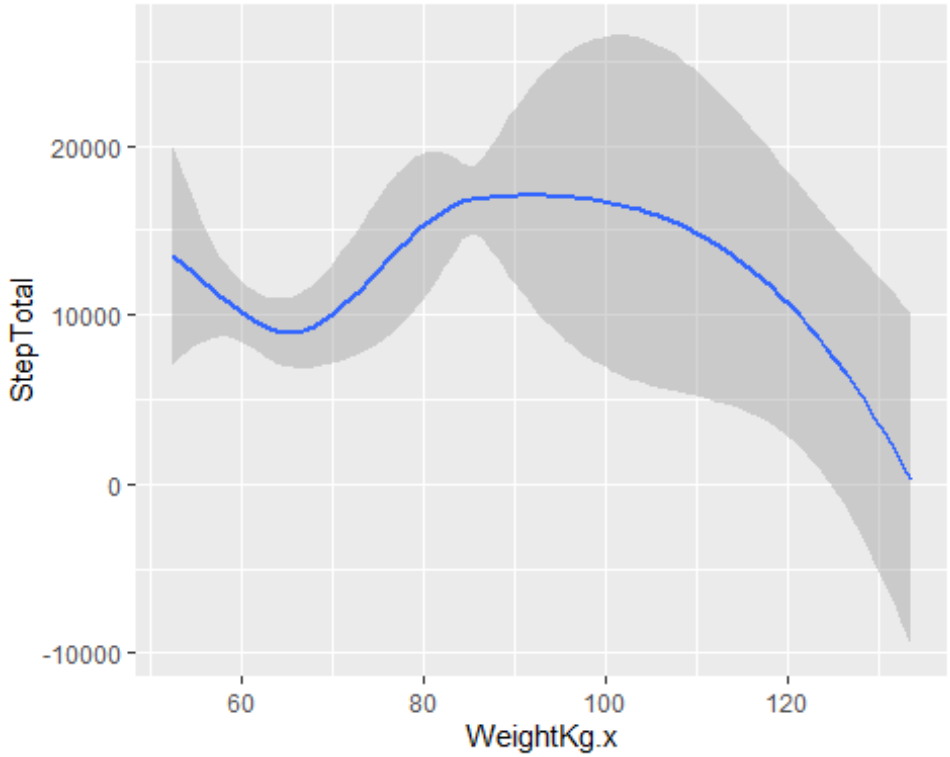



```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = VeryActive
Distance))

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```

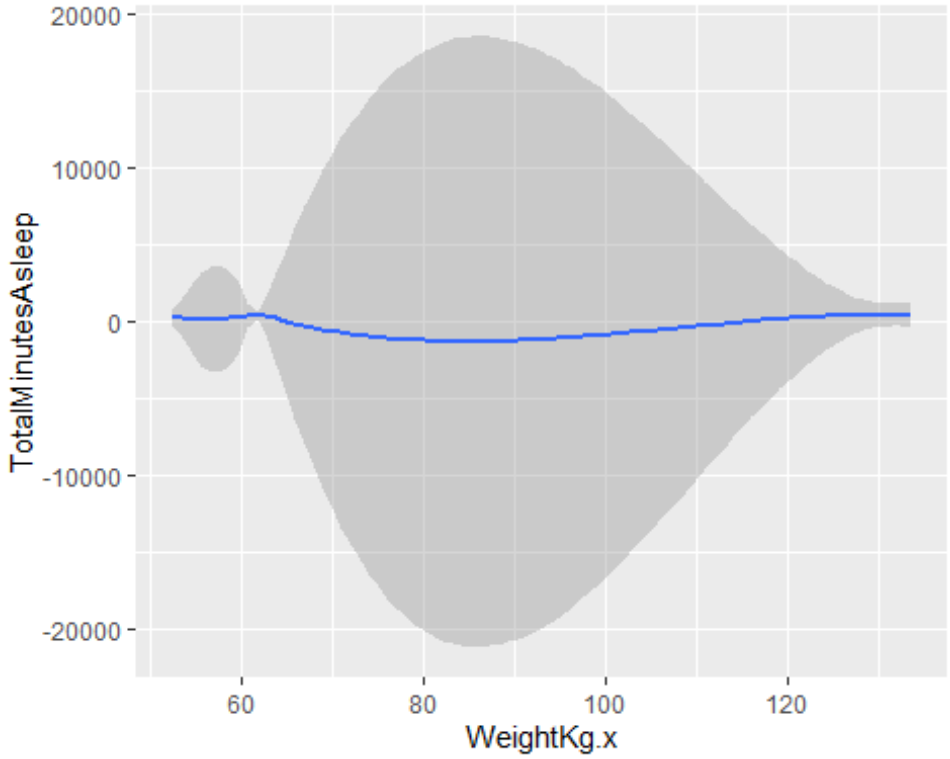


From these plot we can see higher weight people are more sedentary. so we should target specifically below 65kg between 90 and 120kg people are very or fairly active- seems like high weight people are trying to lose weight and exercise more than normal people and they have more very active distance which means they run/jog less and are using indoor activities to stay active such as gym

```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = StepTotal)
)

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

## Warning: Removed 873 rows containing non-finite values (`stat_smooth()`
).
```

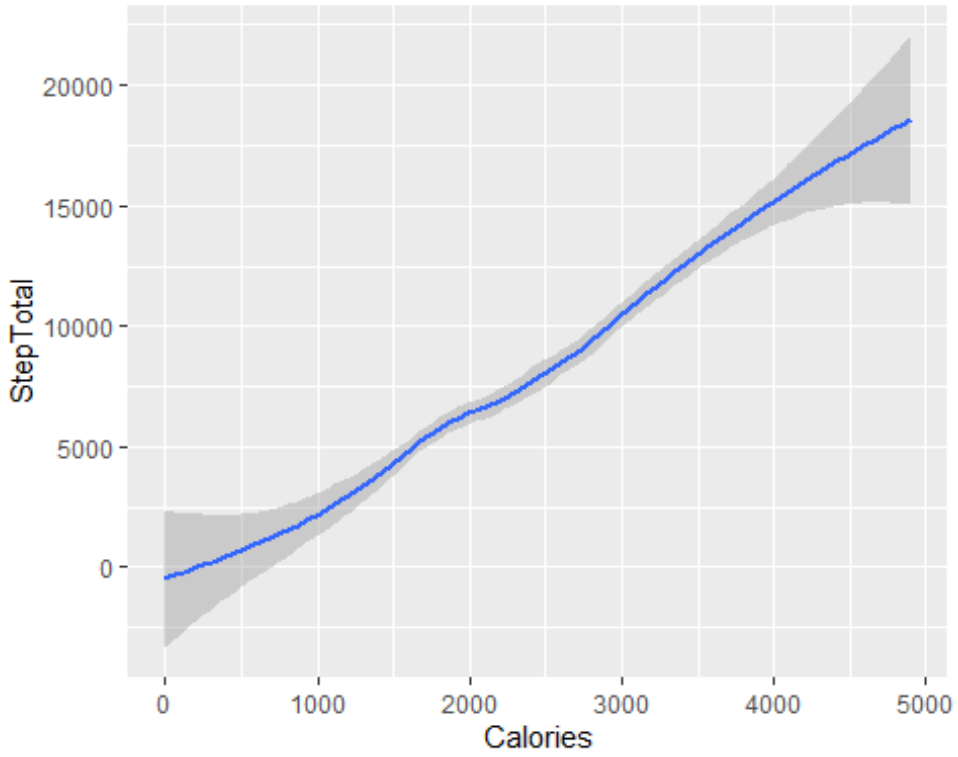



```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=WeightKg.x, y = TotalMinut
esAsleep))

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

## Warning: Removed 905 rows containing non-finite values (`stat_smooth()`
).
```



```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=Calories, y = StepTotal))
```

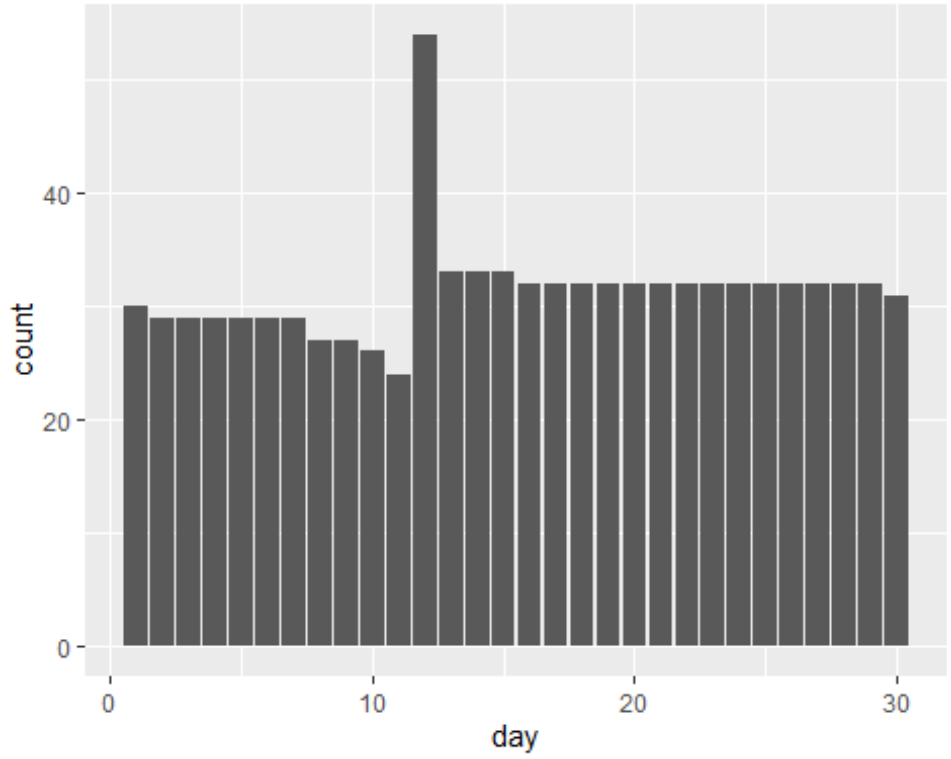
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



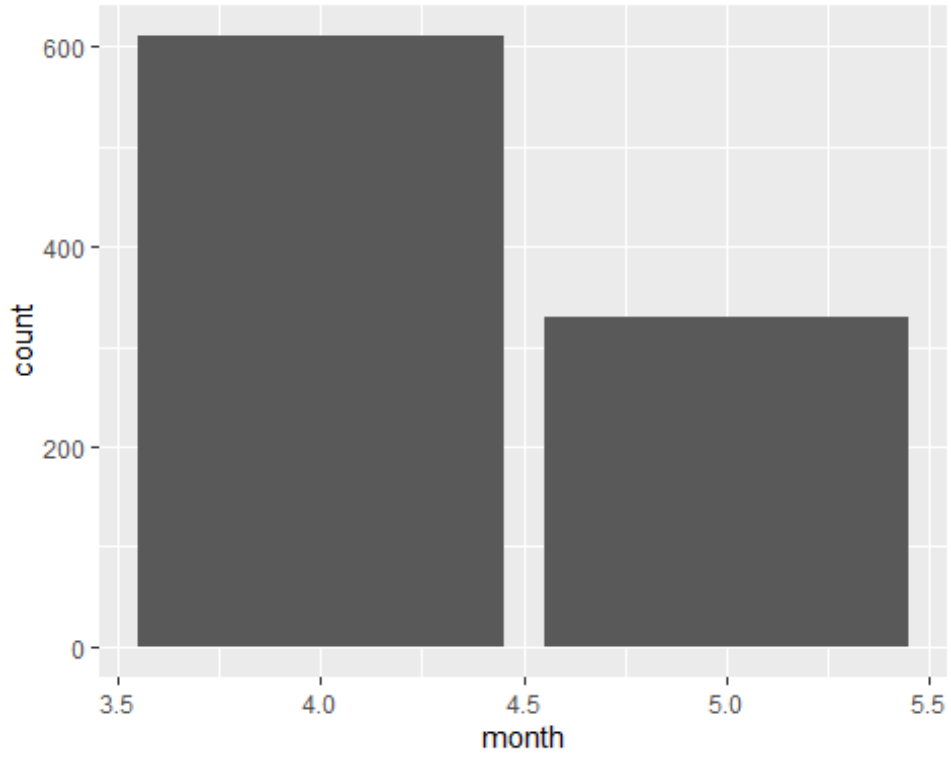
step total decreases with weight, above 100kg steps decline people who take more calories have more steps there is a linear relationship

##Conclusion target less than 65kg as they are active but wont be willing to pay a lot because they are not passionate, they have more active distance though meaning they run/walk more. however between 90 and 120kg people are passionate and would be willing to spend more money

```
ggplot(data=merge_6)+ geom_bar(mapping=aes(x=day))
```



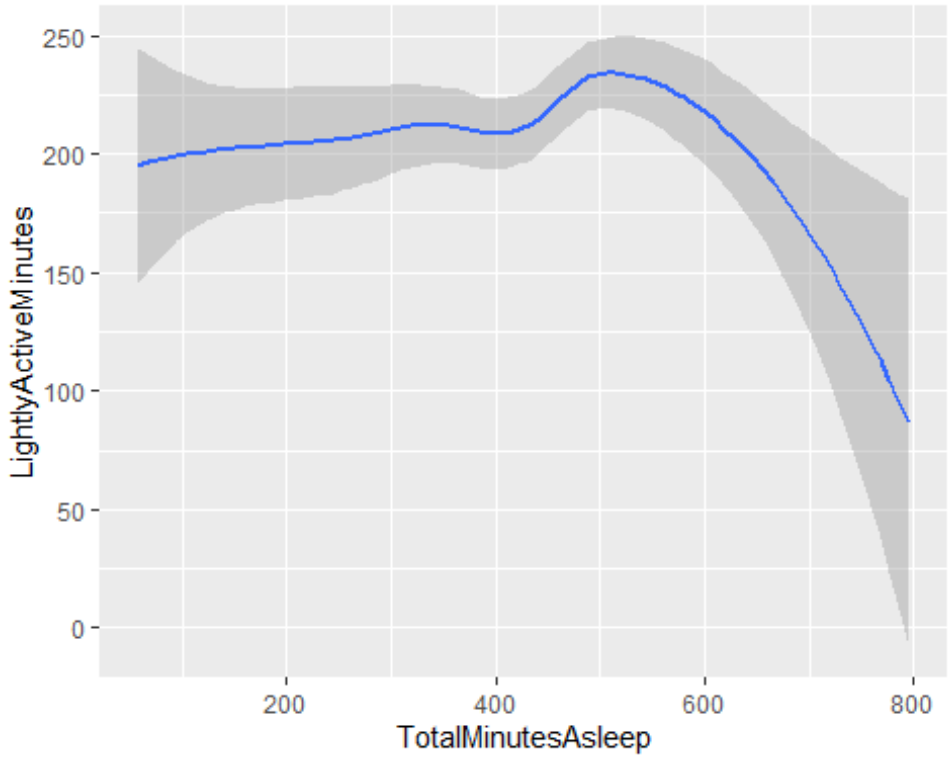

```
ggplot(data=merge_6)+ geom_bar(mapping=aes(x=month))
```



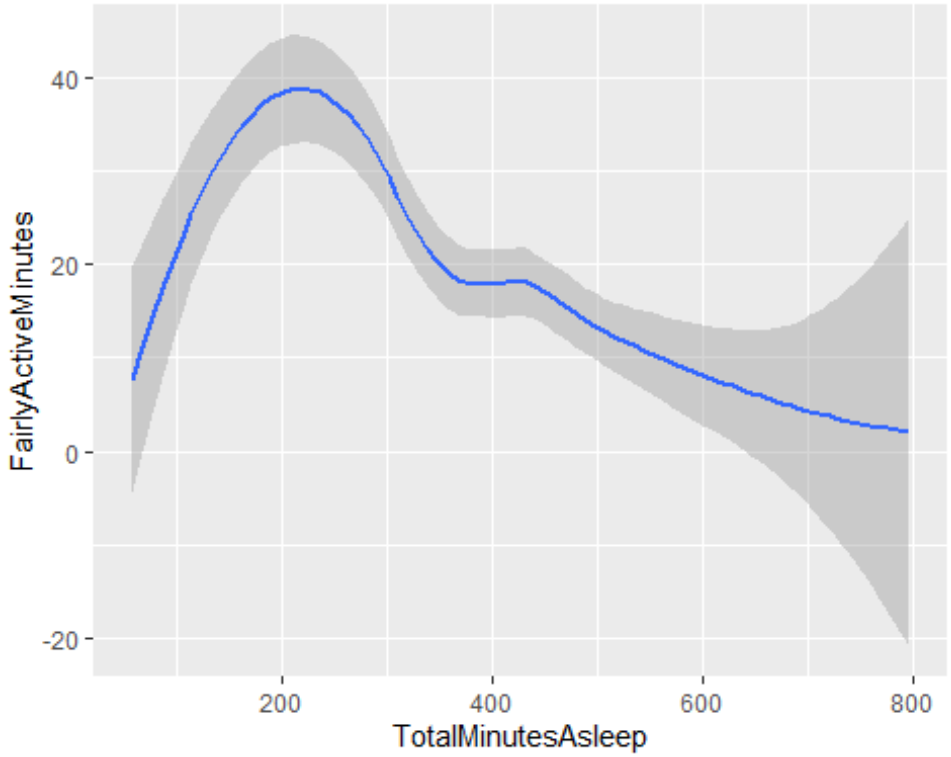
people most active in start of month and middle of month while data was collected only for April and May Months

```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=TotalMinutesAsleep
, y = LightlyActiveMinutes))
```

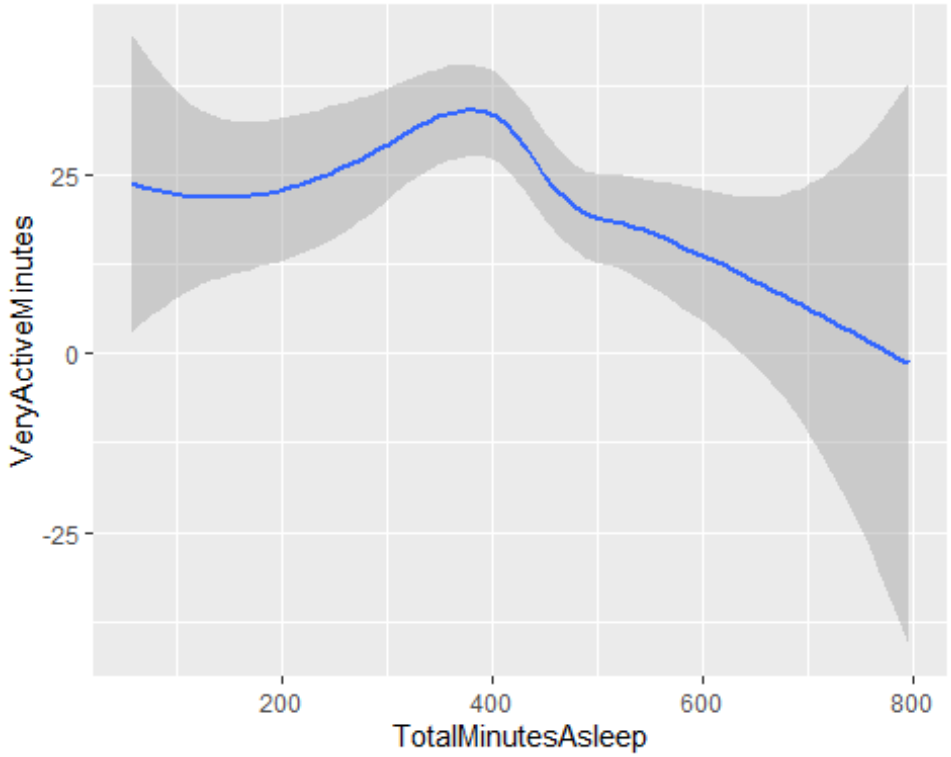
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 530 rows containing non-finite values (`stat_smooth()`
).
```




```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=TotalMinutesAsleep
, y = FairlyActiveMinutes
))

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

## Warning: Removed 530 rows containing non-finite values (`stat_smooth()`
).
```




```
ggplot(data=merge_6) +geom_smooth(mapping=aes(x=TotalMinutesAsleep
, y = VeryActiveMinutes
))

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

## Warning: Removed 530 rows containing non-finite values (`stat_smooth()`
).
```



Active people sleep less than 400 minutes

```
heart_rate <- read.csv("heartrate_seconds_merged.csv")
hourly_calories <- read.csv("hourlyCalories_merged.csv")
hourly_steps <- read.csv("hourlySteps_merged.csv")
```

column names

```
colnames(heart_rate)

## [1] "Id"    "Time"  "Value"

colnames(hourly_calories)

## [1] "Id"           "ActivityHour" "Calories"

colnames(hourly_steps)

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

unique ids

```
n_distinct(heart_rate$Id)

## [1] 14

n_distinct(hourly_calories$Id)

## [1] 33

n_distinct(hourly_steps$Id)

## [1] 33
```

Unique id should be 30 as survey was for 40 people

Format Date and Time

```
heart_rate$Time <- dmy_hms(heart_rate$Time)

## Warning: 1491097 failed to parse.

heart_rate <- heart_rate %>%
  mutate(ActivityDay = as.Date(Time,  format = "%m/%d/%Y"))

heart_rate$Time <- format(as.POSIXct(heart_rate$Time), format = "%H:%M:%S"
)

heart_rate <- heart_rate %>%
  mutate( year = lubridate::year(ActivityDay),
          month = lubridate::month(ActivityDay),
          day = lubridate::day(ActivityDay))

hourly_calories$ActivityHour <- dmy_hms(hourly_calories$ActivityHour)

## Warning: 13821 failed to parse.

hourly_calories <- hourly_calories %>%
  mutate(ActivityDay = as.Date(ActivityHour,  format = "%m/%d/%Y"))

hourly_calories$ActivityHour <- format(as.POSIXct(hourly_calories$Activity
Hour), format = "%H:%M:%S")

hourly_calories <- hourly_calories %>%
  rename(Time = ActivityHour) %>%
  mutate( year = lubridate::year(ActivityDay),
          month = lubridate::month(ActivityDay),
          day = lubridate::day(ActivityDay))

hourly_steps$ActivityHour <- dmy_hms(hourly_steps$ActivityHour)

## Warning: 13821 failed to parse.

hourly_steps <- hourly_steps %>%
  mutate(ActivityDay = as.Date(ActivityHour,  format = "%m/%d/%Y"))

hourly_steps$ActivityHour <- format(as.POSIXct(hourly_steps$ActivityHour),
format = "%H:%M:%S")

hourly_steps <- hourly_steps %>%
  rename(Time = ActivityHour)%>%
  mutate( year = lubridate::year(ActivityDay),
          month = lubridate::month(ActivityDay),
          day = lubridate::day(ActivityDay))
```

Plot Graphs

```
average_values_2 <- data.frame(avg_heart_rate =mean(heart_rate$Value),

avg_hourly_steps = mean(hourly_steps$StepTotal),
```

```
avg_hourly_calories = mean(hourly_calories$Calories))

average_values <- merge(average_values1,average_values_2)

head(average_values)
```

```
##   avg_sedentary_mins avg_light_active_mins avg_fairly_active_mins
## 1           991.2106              192.8128               13.56489
##   avg_very_active_mins avg_sedentary_active_distance avg_light_active_d
istance
## 1             21.16489                   0.001606383
3.34
##   avg_moderately_active_distance avg_very_active_distance avg_minutes_s
leep
## 1                      0.5675426                1.502681          419.
1732
##   avg_calories avg_steps avg_heart_rate avg_hourly_steps avg_hourly_cal
ories
## 1      2303.61  7637.911       77.32842         320.1663            97.
38676
```

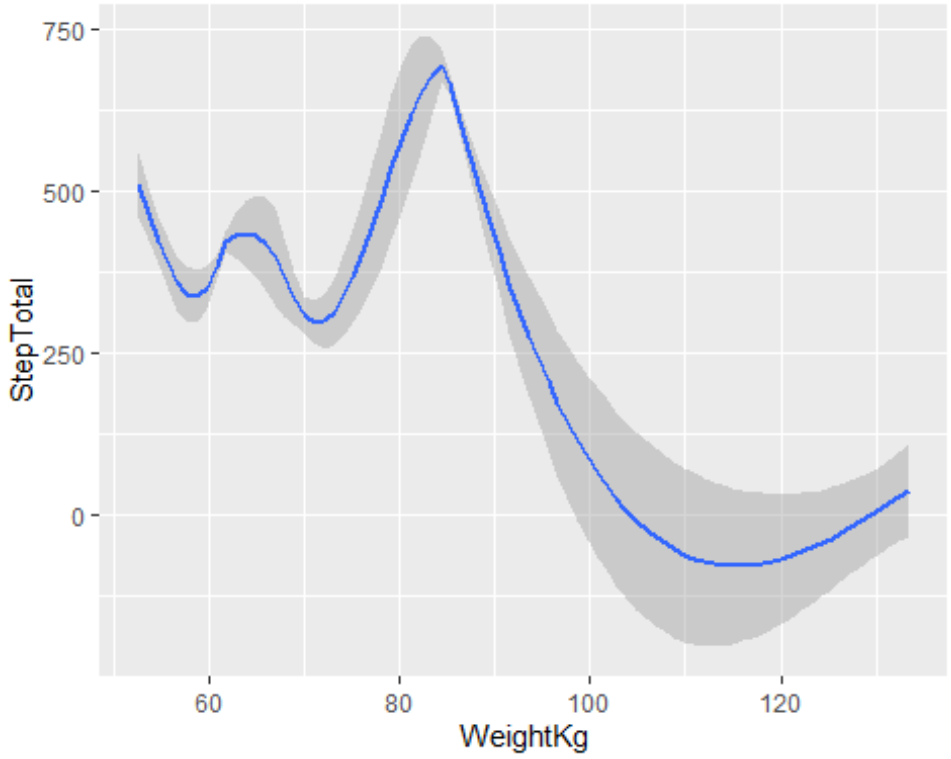
```
setwd("Created DataSets")
export(average_values, "Complete Average Values.xlsx")
```

merge data

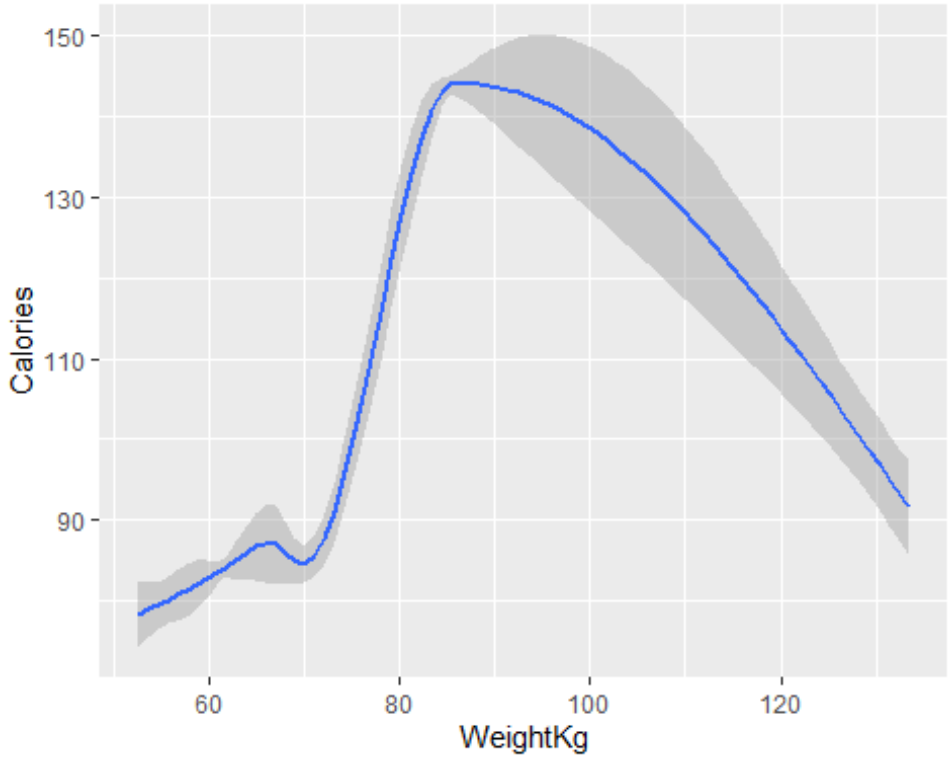
```
merge_7 <- merge(daily_activity, heart_rate, by = c("Id"))

merge_8 <- merge(daily_activity, hourly_calories, by = c("Id"))
merge_9 <- merge(daily_activity, hourly_steps, by = c("Id"))
```

Plot Graphs

```
ggplot(data=merge_9) +geom_smooth(mapping=aes(x=WeightKg, y = StepTotal))

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




```
ggplot(data=merge_8) +geom_smooth(mapping=aes(x=WeightKg, y = Calories))

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



People between 80 and 100kg take most calories

4. Conclusion

• We can see higher weight people (above 120kg) are more sedentary. so we should target specifically below 65kg but between 90 and 120kg people are very or fairly active - seems like high weight people are trying to lose weight and exercise more than normal people but they have less (very active distance) which means they run/jog less and are using indoor activities to stay active such as gym

• less than 70kg as they are active but won't be willing to pay a lot because they are not passionate, they have more active distance though meaning they run/walk more. however, between 90 and 120kg people are passionate and would be willing to spend more money

• People are most active in the start of month and middle of month while data only collected for April and May.

• Step total decreases with weight, above 100kg steps decline. People who take more calories have more steps there is a linear relationship, while active people sleep less than 400 minutes and people between 80 and 100kg take most calories.

• Bellabeat's marketing team can encourage users by educating and equipping them with knowledge about fitness benefits, suggest different types of exercises, calories intake and burn rate information on Bellabeat's application.

• Most people use fitbit to track steps and calories burned, people don't use to track sleep much. I will suggest focusing on steps, calories more than sleep in application

• The relation between steps taken vs calories burned and very active minutes vs calories burned shows positive correlation. So, this can be a good marketing strategy.

• If users want to lose weight, it's probably a good idea to control daily calorie consumption. Bellabeat's can suggest some ideas for low-calorie lunch and dinner.

• The Bellabeat app can recommend reducing sedentary time.