# Reproducable Research, Project 1

## Part 1: Loading and preprocessing the data

Data found here: <https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip> Download data and unzip files into your working direcory Next, Load the data into R:

setwd("C:/Users/LindseyBehrens/OneDrive - Genective/Documents/R Working/Coursera")  
  
Data <- read.csv("activity.csv")  
head(Data)

## steps date interval  
## 1 NA 2012-10-01 0  
## 2 NA 2012-10-01 5  
## 3 NA 2012-10-01 10  
## 4 NA 2012-10-01 15  
## 5 NA 2012-10-01 20  
## 6 NA 2012-10-01 25

Take a look at the Data:

str(Data)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : chr "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

Notice that the ‘date’ column is read as a character instead of a date. Let’s fix that:

Data$date <- as.Date(Data$date)  
str(Data)

## 'data.frame': 17568 obs. of 3 variables:  
## $ steps : int NA NA NA NA NA NA NA NA NA NA ...  
## $ date : Date, format: "2012-10-01" "2012-10-01" ...  
## $ interval: int 0 5 10 15 20 25 30 35 40 45 ...

## Part 2: What is mean total number of steps taken per day?

We will ignore the missing values here. Lets fine the total number of steps:

totalSteps <- sum(Data$steps, na.rm = TRUE)  
totalSteps

## [1] 570608

570608 total steps!

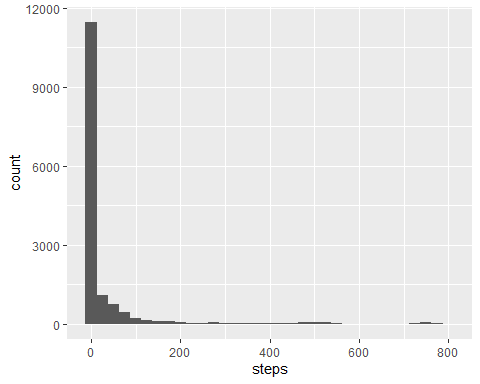
Lets look at the distribution in a histogram:

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.2.3

Histogram <- ggplot(Data, aes(x=steps)) +  
 geom\_histogram(binwidth=25)  
Histogram

## Warning: Removed 2304 rows containing non-finite values (`stat\_bin()`).



That’s a lot of 0 steps.

Lets look at the mean and median number of steps:

meanSteps <- mean(Data$steps, na.rm = TRUE)  
meanSteps

## [1] 37.3826

medianSteps <- median(Data$steps, na.rm = TRUE)  
medianSteps

## [1] 0

Using this code we found that the mean number of steps was 37.3826 and the median was 0 steps.

## Part 3: What is the average daily activity pattern?

Lets make a graph to look at the average number of steps in each 5-minute interval of the day

library(dplyr)

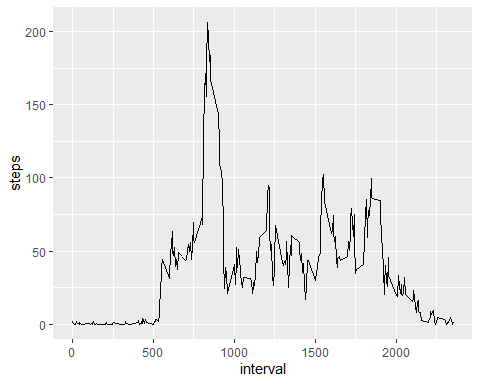
## Warning: package 'dplyr' was built under R version 4.2.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

Averaged <- Data %>%  
 group\_by(interval) %>%  
 summarize(steps = mean(steps, na.rm = TRUE))  
  
  
AverageSteps <- ggplot(Averaged, aes(y=steps, x= interval)) +  
 geom\_line()  
AverageSteps



Now lets find the interval with the highest average number of setps

Highest <- seq(along=Averaged$steps)[Averaged$steps==max(Averaged$steps)]  
Averaged[Highest,]

## # A tibble: 1 × 2  
## interval steps  
## <int> <dbl>  
## 1 835 206.

Interval 835 with an average of 206 steps!

## Part 4: Imputing missing values

Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

Find the total number of rows with missing values:

Data\_No\_NA <- Data[complete.cases(Data), ]  
NA\_count <- nrow(Data) - nrow(Data\_No\_NA)  
NA\_count

## [1] 2304

There are 2304 missing values

Now we will fill the missing NA’s with the median

library(dplyr)  
library(tidyr)

## Warning: package 'tidyr' was built under R version 4.2.3

Imputed\_Data <- Data %>%   
 mutate(steps = replace\_na(steps,median(steps, na.rm = TRUE)))  
nrow(Imputed\_Data) == nrow(Data)

## [1] TRUE

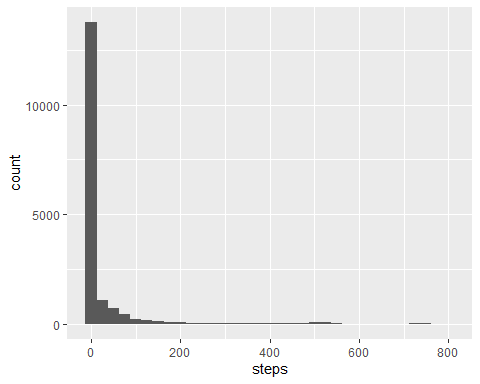
head(Imputed\_Data)

## steps date interval  
## 1 0 2012-10-01 0  
## 2 0 2012-10-01 5  
## 3 0 2012-10-01 10  
## 4 0 2012-10-01 15  
## 5 0 2012-10-01 20  
## 6 0 2012-10-01 25

After replacing the NA’s with the median the data sets are the same length and there are no more NA’s

Now Lets make a histogram of the total number of steps taken each day

Histogram2 <- ggplot(Imputed\_Data, aes(x=steps)) +  
 geom\_histogram(binwidth=25)  
Histogram2



And look at the new mean and median

meanSteps2 <- mean(Imputed\_Data$steps, na.rm = TRUE)  
meanSteps2

## [1] 32.47996

medianSteps2 <- median(Imputed\_Data$steps, na.rm = TRUE)  
medianSteps2

## [1] 0

Do these values differ from the estimates from the first part of the assignment? The mean is slightly lower and the median is the same.

meanSteps

## [1] 37.3826

meanSteps2

## [1] 32.47996

medianSteps

## [1] 0

medianSteps2

## [1] 0

What is the impact of imputing missing data on the estimates of the total daily number of steps?

totalSteps2 <- sum(Imputed\_Data$steps, na.rm = TRUE)  
totalSteps2

## [1] 570608

totalSteps

## [1] 570608

The same! The median was 0.

## Part 5: Are there differences in activity patterns between weekdays and weekends?

Create a new factor variable in the dataset with two levels indicating whether a given date is a weekday or weekend day.

Imputed\_Data$Weekday <- weekdays(Imputed\_Data$date)  
Day <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")  
End <- c("Saturday", "Sunday")  
  
Imputed\_Data$Weekend <- Imputed\_Data$Weekday %in% End  
Imputed\_Data <- Imputed\_Data %>%  
 mutate(DayClass = case\_when(  
 Weekend == TRUE ~ "Weekend",  
 Weekend == FALSE ~ "Weekday"  
 ))  
  
head(Imputed\_Data)

## steps date interval Weekday Weekend DayClass  
## 1 0 2012-10-01 0 Monday FALSE Weekday  
## 2 0 2012-10-01 5 Monday FALSE Weekday  
## 3 0 2012-10-01 10 Monday FALSE Weekday  
## 4 0 2012-10-01 15 Monday FALSE Weekday  
## 5 0 2012-10-01 20 Monday FALSE Weekday  
## 6 0 2012-10-01 25 Monday FALSE Weekday

Make a time series plot of the 5-minute interval and the average number of steps taken, averaged across all weekday days or weekend days

Classes <- unique(Imputed\_Data$DayClass)  
Averaged2 <- data.frame(matrix(ncol=3, nrow=0))  
for (Class in Classes) {  
 set <- subset(Imputed\_Data, DayClass == Class)  
 Average <- set %>%  
 group\_by(interval) %>%  
 summarize(steps = mean(steps, na.rm = TRUE))  
 Average$DayClass <- Class  
 Averaged2 <- rbind(Averaged2, Average)  
}  
  
AverageSteps2 <- ggplot(Averaged2, aes(y=steps, x= interval)) +  
 geom\_line() +  
 facet\_grid( ~ DayClass)  
AverageSteps2

