Reproducible Research Week 2 Project 1

Renga Pandurengan

6/30/2021

Loading required packages

library(readr)  
library(dplyr)

##   
## 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

library(ggplot2)  
library(Hmisc)

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following objects are masked from 'package:base':  
##   
## format.pval, units

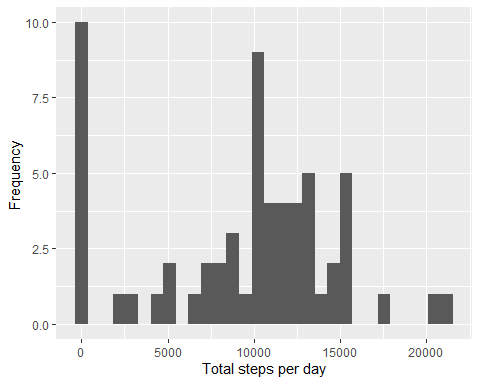
1. Loading & Manipulating data

data <- read.csv("C:/Users/rkrishnaramanujam/Desktop/Data\_Science/ReproducibleResearch/Week\_2/Project\_1/repdata\_data\_activity/activity.csv")  
data$date\_new <- as.Date(data$date)

1. Creating Histogram of the total number of steps taken each day

stepsPerDay <- tapply(data$steps, data$date, sum, na.rm=TRUE)  
qplot(stepsPerDay, xlab='Total steps per day', ylab='Frequency ')

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

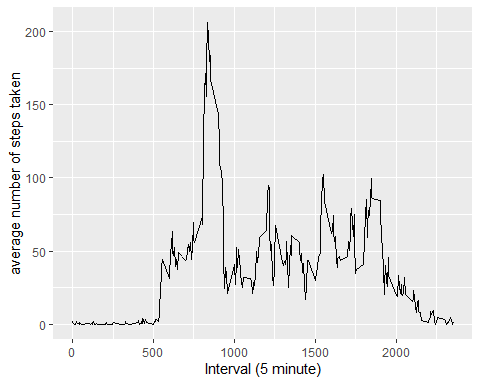


1. Mean and median number of steps taken each day

MeanstepsByDay <- mean(stepsPerDay)  
MedianstepsByDay <- median(stepsPerDay)

1. Time series plot of the average number of steps taken

averageStepsPerTimeBlock <- aggregate(x=list(meanSteps=data$steps), by=list(interval=data$interval), FUN=mean, na.rm=TRUE)  
  
  
ggplot(data=averageStepsPerTimeBlock, aes(x=interval, y=meanSteps)) +  
 geom\_line() +  
 xlab("Interval (5 minute)") +  
 ylab("average number of steps taken")



1. The 5-minute interval that, on average, contains the maximum number of steps

maxSteps <- which.max(averageStepsPerTimeBlock$meanSteps)  
timeMaxSteps <- gsub("([0-9]{1,2})([0-9]{2})", "\\1:\\2", averageStepsPerTimeBlock[maxSteps,'interval'])

1. Code to describe and show a strategy for imputing missing data

#Number of missing values  
NoMissingValues <- sum(is.na(data$steps))  
NoMissingValues

## [1] 2304

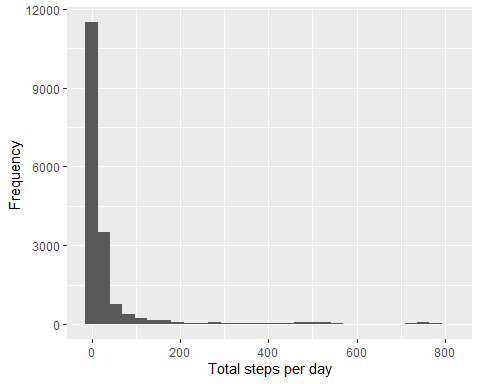
#Strategy for imputing missing values  
dataImputed <- data  
dataImputed$steps <- impute(data$steps, fun=mean)

1. Histogram of the total number of steps taken each day after missing values are imputed

# Create the histogram with imputed data.  
qplot(dataImputed$steps, xlab='Total steps per day', ylab='Frequency ')

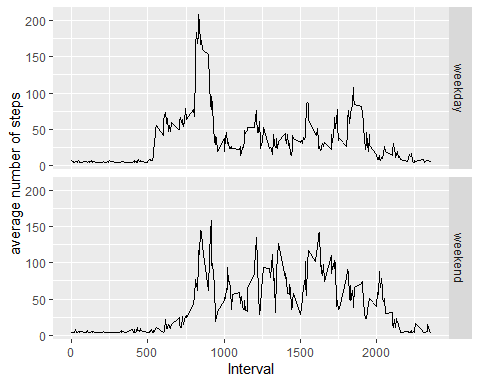
## Don't know how to automatically pick scale for object of type impute. Defaulting to continuous.

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



1. Panel plot comparing the average number of steps taken per 5-minute interval across weekdays and weekends

dataImputed$dateType <- ifelse(as.POSIXlt(dataImputed$date)$wday %in% c(0,6), 'weekend', 'weekday')  
MeandataImputed <- aggregate(steps ~ interval + dateType, data=dataImputed, mean)  
  
# Panel plot   
  
ggplot(MeandataImputed, aes(interval, steps)) +   
 geom\_line() +   
 facet\_grid(dateType ~ .) +  
 xlab("Interval") +   
 ylab("average number of steps")



1. All of the R code needed to reproduce the results (numbers, plots, etc.) in the report