|  |
| --- |
| title: “Reproducible Research Project 1” |
| author: “Jaime Espinoza” |
| date: “23/05/2022” |
| output: |
| md\_document: |
| variant: markdown\_github |

github repo for rest of specialization: [Data Science Coursera](https://github.com/jcespinoza2022/datasciencecoursera)

## Introduction

Hoy en día es posible recopilar una gran cantidad de datos sobre el movimiento personal mediante dispositivos de monitorización de la actividad como Fitbit, Nike Fuelband o Jawbone Up. Este tipo de dispositivos forman parte del movimiento del “yo cuantificado”, un grupo de entusiastas que toman mediciones sobre sí mismos con regularidad para mejorar su salud, para encontrar patrones en su comportamiento o porque son frikis de la tecnología. Pero estos datos siguen estando infrautilizados, tanto porque los datos brutos son difíciles de obtener como por la falta de métodos estadísticos y de software para procesar e interpretar los datos.

Esta tarea utiliza los datos de un dispositivo de monitorización de la actividad personal. Este dispositivo recoge datos a intervalos de 5 minutos a lo largo del día. Los datos consisten en dos meses de datos de un individuo anónimo recogidos durante los meses de octubre y noviembre de 2012 e incluyen el número de pasos dados en intervalos de 5 minutos cada día.

Los datos para esta tarea pueden descargarse del sitio web del curso:

Dataset: [Activity monitoring data](https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip)

Las variables incluidas en este conjunto de datos son:

pasos : Número de pasos dados en un intervalo de 5 minutos (los valores faltantes se codifican comoN / A) date : La fecha en la que se tomó la medida en formato AAAA-MM-DD intervalo : Identificador del intervalo de 5 minutos en el que se tomó la medición El conjunto de datos se almacena en un archivo de valores separados por comas (CSV) y hay un total de 17 568 observaciones en este conjunto de datos.

## Carga y preprocesamiento de los datos

Descomprimir los datos para obtener un archivo csv.

library("data.table")  
library(ggplot2)  
  
fileUrl <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2Factivity.zip"  
download.file(fileUrl, destfile = paste0(getwd(), '/repdata%2Fdata%2Factivity.zip'), method = "curl")  
unzip("repdata%2Fdata%2Factivity.zip",exdir = "data")

##Lectura de datos csv en Data.Table

activityDT <- data.table::fread(input = "data/activity.csv")

## ¿Cuál es la media del número total de pasos dados al día?

1. Calculate the total number of steps taken per day

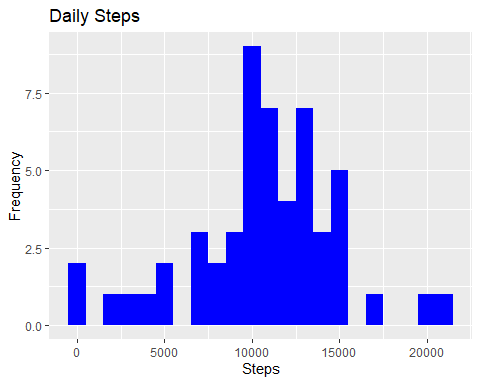
Total\_Steps <- activityDT[, c(lapply(.SD, sum, na.rm = FALSE)), .SDcols = c("steps"), by = .(date)]   
  
head(Total\_Steps, 10)

## date steps  
## 1: 2012-10-01 NA  
## 2: 2012-10-02 126  
## 3: 2012-10-03 11352  
## 4: 2012-10-04 12116  
## 5: 2012-10-05 13294  
## 6: 2012-10-06 15420  
## 7: 2012-10-07 11015  
## 8: 2012-10-08 NA  
## 9: 2012-10-09 12811  
## 10: 2012-10-10 9900

1. Haz un histograma del número total de pasos dados cada día.

ggplot(Total\_Steps, aes(x = steps)) +  
 geom\_histogram(fill = "blue", binwidth = 1000) +  
 labs(title = "Daily Steps", x = "Steps", y = "Frequency")

## Warning: Removed 8 rows containing non-finite values (stat\_bin).



1. Calcular y comunicar la media y la mediana del número total de pasos dados al día

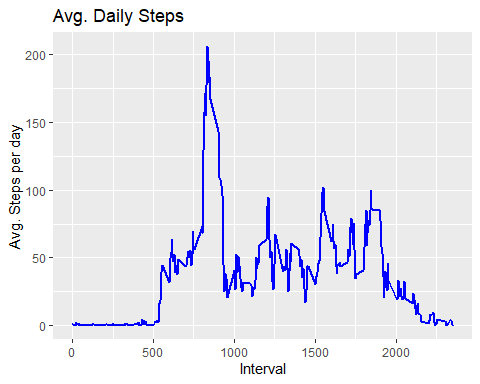
Total\_Steps[, .(Mean\_Steps = mean(steps, na.rm = TRUE), Median\_Steps = median(steps, na.rm = TRUE))]

## Mean\_Steps Median\_Steps  
## 1: 10766.19 10765

##¿Cuál es el patrón medio de actividad diaria?

1. Realiza un gráfico de series temporales 𝚝𝚢𝚙𝚎 = “𝚕” del intervalo de 5 minutos (eje x) y el número medio de pasos dados, promediado en todos los días (eje y)

IntervalDT <- activityDT[, c(lapply(.SD, mean, na.rm = TRUE)), .SDcols = c("steps"), by = .(interval)]   
  
ggplot(IntervalDT, aes(x = interval , y = steps)) + geom\_line(color="blue", size=1) + labs(title = "Avg. Daily Steps", x = "Interval", y = "Avg. Steps per day")



1. ¿Qué intervalo de 5 minutos, de media en todos los días del conjunto de datos, contiene el máximo número de pasos?

IntervalDT[steps == max(steps), .(max\_interval = interval)]

## max\_interval  
## 1: 835

## Imputar valores perdidos

1. Calcule y comunique el número total de valores perdidos en el conjunto de datos (i.e. el número total de filas con 𝙽𝙰s)

activityDT[is.na(steps), .N ]

## [1] 2304

# solución alternativa  
  
nrow(activityDT[is.na(steps),])

## [1] 2304

1. Diseñe una estrategia para rellenar todos los valores que faltan en el conjunto de datos. La estrategia no tiene por qué ser sofisticada. Por ejemplo, puede utilizar la media/mediana de ese día, o la media de ese intervalo de 5 minutos, etc.

# Filling in missing values with median of dataset.   
activityDT[is.na(steps), "steps"] <- activityDT[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("steps")]

1. Cree un nuevo conjunto de datos que sea igual al conjunto de datos original pero con los datos que faltan rellenados.

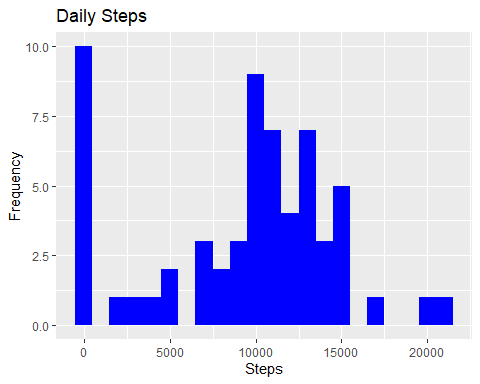
data.table::fwrite(x = activityDT, file = "data/tidyData.csv", quote = FALSE)

1. Haz un histograma del número total de pasos dados cada día y calcula y comunica la media y la mediana del número total de pasos dados al día. ¿Difieren estos valores de las estimaciones de la primera parte de la tarea? ¿Cuál es el impacto de la imputación de los datos que faltan en las estimaciones del número total de pasos diarios?

# Número total de pasos dados al día  
Total\_Steps <- activityDT[, c(lapply(.SD, sum)), .SDcols = c("steps"), by = .(date)]   
  
#media y mediana del número total de pasos dados al día  
Total\_Steps[, .(Mean\_Steps = mean(steps), Median\_Steps = median(steps))]

## Mean\_Steps Median\_Steps  
## 1: 9354.23 10395

ggplot(Total\_Steps, aes(x = steps)) + geom\_histogram(fill = "blue", binwidth = 1000) + labs(title = "Daily Steps", x = "Steps", y = "Frequency")



| Tipo de estimacion | median\_Steps | Mediana\_Steps |
| --- | --- | --- |
| primera Parte (con na) | 10765 | 10765 |
| Second Part (rellenar na con la mediana) | 9354.23 | 10395 |

## ¿Hay diferencias en los patrones de actividad entre los días de semana y los fines de semana?

1.Cree una nueva variable de factor en el conjunto de datos con dos niveles: “día de la semana” y “fin de semana” que indique si una fecha determinada es un día de la semana o del fin de semana.

# JSolo hay que recrear la actividadDT desde cero y luego hacer el nuevo factor variable. (No es necesario, sólo quiero tener claro cuál es el proceso completo).   
activityDT <- data.table::fread(input = "data/activity.csv")  
activityDT[, date := as.POSIXct(date, format = "%Y-%m-%d")]  
activityDT[, `Day of Week`:= weekdays(x = date)]  
activityDT[grepl(pattern = "Monday|Tuesday|Wednesday|Thursday|Friday", x = "Day of Week"), "weekday or weekend"] <- "weekday"  
activityDT[grepl(pattern = "Saturday|Sunday", x = "Day of Week"), "weekday or weekend"] <- "weekend"  
activityDT[, `weekday or weekend` := as.factor("weekday or weekend")]  
head(activityDT, 10)

## steps date interval Day of Week weekday or weekend  
## 1: NA 2012-10-01 0 lunes weekday or weekend  
## 2: NA 2012-10-01 5 lunes weekday or weekend  
## 3: NA 2012-10-01 10 lunes weekday or weekend  
## 4: NA 2012-10-01 15 lunes weekday or weekend  
## 5: NA 2012-10-01 20 lunes weekday or weekend  
## 6: NA 2012-10-01 25 lunes weekday or weekend  
## 7: NA 2012-10-01 30 lunes weekday or weekend  
## 8: NA 2012-10-01 35 lunes weekday or weekend  
## 9: NA 2012-10-01 40 lunes weekday or weekend  
## 10: NA 2012-10-01 45 lunes weekday or weekend

1. Haga un gráfico de panel que contenga un gráfico de serie temporal (es decir, 𝚝𝚢𝚙𝚎 = “𝚕”) del intervalo de 5 minutos (eje x) y el número medio de pasos dados, promediado en todos los días de la semana o del fin de semana (eje y). Consulte el archivo README en el repositorio de GitHub para ver un ejemplo de cómo debería ser este gráfico utilizando datos simulados.

activityDT[is.na(steps), "steps"] <- activityDT[, c(lapply(.SD, median, na.rm = TRUE)), .SDcols = c("steps")]  
IntervalDT <- activityDT[, c(lapply(.SD, mean, na.rm = TRUE)), .SDcols = c("steps"), by = .(interval)]  
  
ggplot(IntervalDT , aes(x = interval , y = steps, color="weekday or weekend")) + geom\_line() + labs(title = "Avg. Daily Steps by Weektype", x = "Interval", y = "No. of Steps") + facet\_wrap(~'weekday or weekend' , ncol = 1, nrow=2)

