Reproducible Research: Peer Assessment 1

Felipe Villota

Creating an setting the directory

```
dir()
```

```
## [1] "activity.csv"                  "activity.zip"
## [3] "doc"                           "instructions_fig"
## [5] "PA1_template.Rmd"              "R5P1"
## [7] "README.md"                     "RepData_PeerAssessment1.Rproj"
```

```
if(!file.exists("R5P1")){dir.create("R5P1")}
dir("R5P1")
```

```
## character(0)
```

```
setwd("C:/Users/USER/Desktop/R/R5P1")
```

Load and explore CSV file

```
DATA<- read.csv("activity.csv")
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 ...
```

```
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
```

```r
dim(DATA)
```

```
## [1] 17568     3
```

```r
names(DATA)
```

```
## [1] "steps"    "date"     "interval"
```

1. What is mean total number of steps taken per day?

Total number of steps taken per day

```r
stepsperday <- aggregate(DATA$steps, list(DATA$date), FUN=sum)
colnames(stepsperday) <- c("Date", "Steps")
stepsperday
```

```
##          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
## 11 2012-10-11 10304
## 12 2012-10-12 17382
## 13 2012-10-13 12426
## 14 2012-10-14 15098
## 15 2012-10-15 10139
## 16 2012-10-16 15084
## 17 2012-10-17 13452
## 18 2012-10-18 10056
## 19 2012-10-19 11829
## 20 2012-10-20 10395
## 21 2012-10-21  8821
## 22 2012-10-22 13460
## 23 2012-10-23  8918
## 24 2012-10-24  8355
## 25 2012-10-25  2492
## 26 2012-10-26  6778
## 27 2012-10-27 10119
## 28 2012-10-28 11458
## 29 2012-10-29  5018
## 30 2012-10-30  9819
## 31 2012-10-31 15414
## 32 2012-11-01    NA
```

```
## 33 2012-11-02 10600
## 34 2012-11-03 10571
## 35 2012-11-04    NA
## 36 2012-11-05 10439
## 37 2012-11-06  8334
## 38 2012-11-07 12883
## 39 2012-11-08  3219
## 40 2012-11-09    NA
## 41 2012-11-10    NA
## 42 2012-11-11 12608
## 43 2012-11-12 10765
## 44 2012-11-13  7336
## 45 2012-11-14    NA
## 46 2012-11-15    41
## 47 2012-11-16  5441
## 48 2012-11-17 14339
## 49 2012-11-18 15110
## 50 2012-11-19  8841
## 51 2012-11-20  4472
## 52 2012-11-21 12787
## 53 2012-11-22 20427
## 54 2012-11-23 21194
## 55 2012-11-24 14478
## 56 2012-11-25 11834
## 57 2012-11-26 11162
## 58 2012-11-27 13646
## 59 2012-11-28 10183
## 60 2012-11-29  7047
## 61 2012-11-30    NA
```

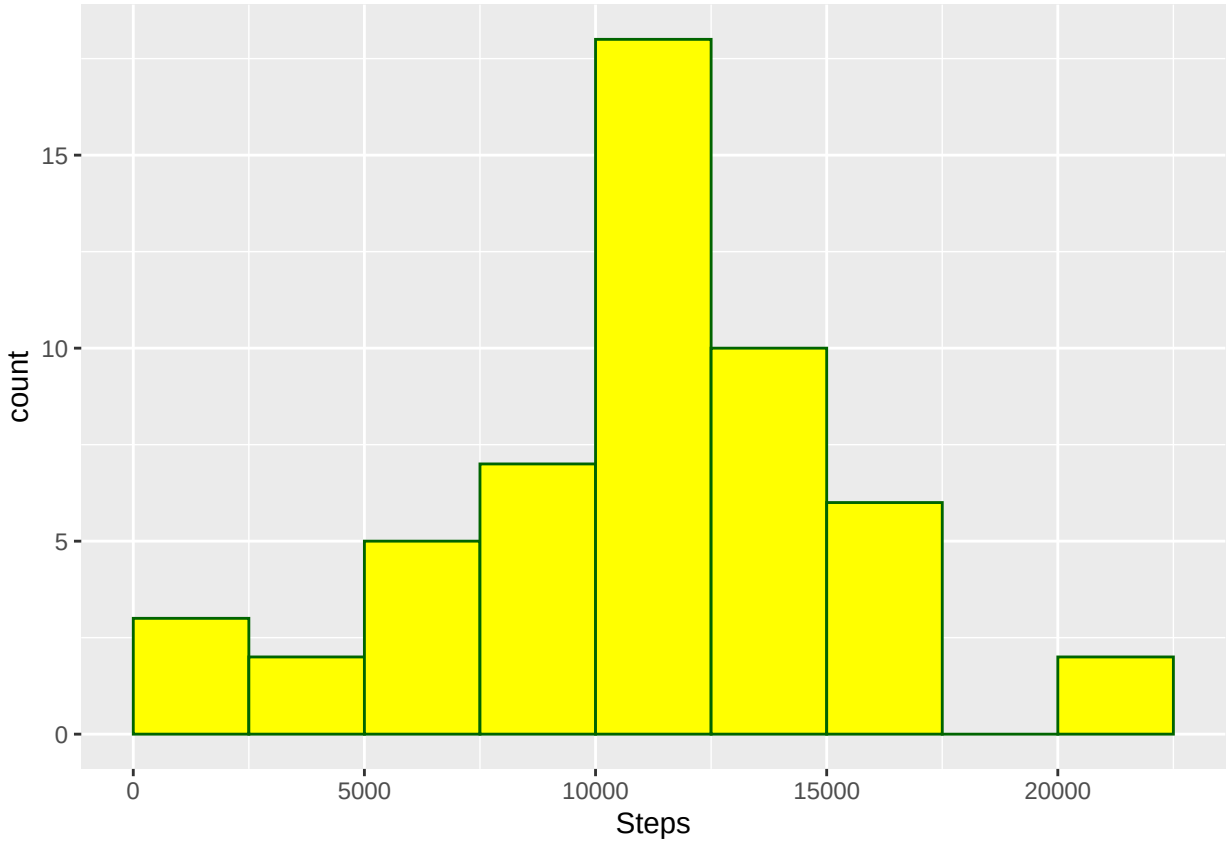
Histogram of the total number of steps taken each day

```
library(ggplot2)
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
```

```
stephist <- ggplot(stepsperday, aes(Steps))
stephist+geom_histogram(boundary=0, binwidth=2500, col="darkgreen",
        fill="yellow")
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



Mean and median of the total number of steps taken per day

```
mean(stepsperday$Steps, na.rm=TRUE)
```

```
## [1] 10766.19
```

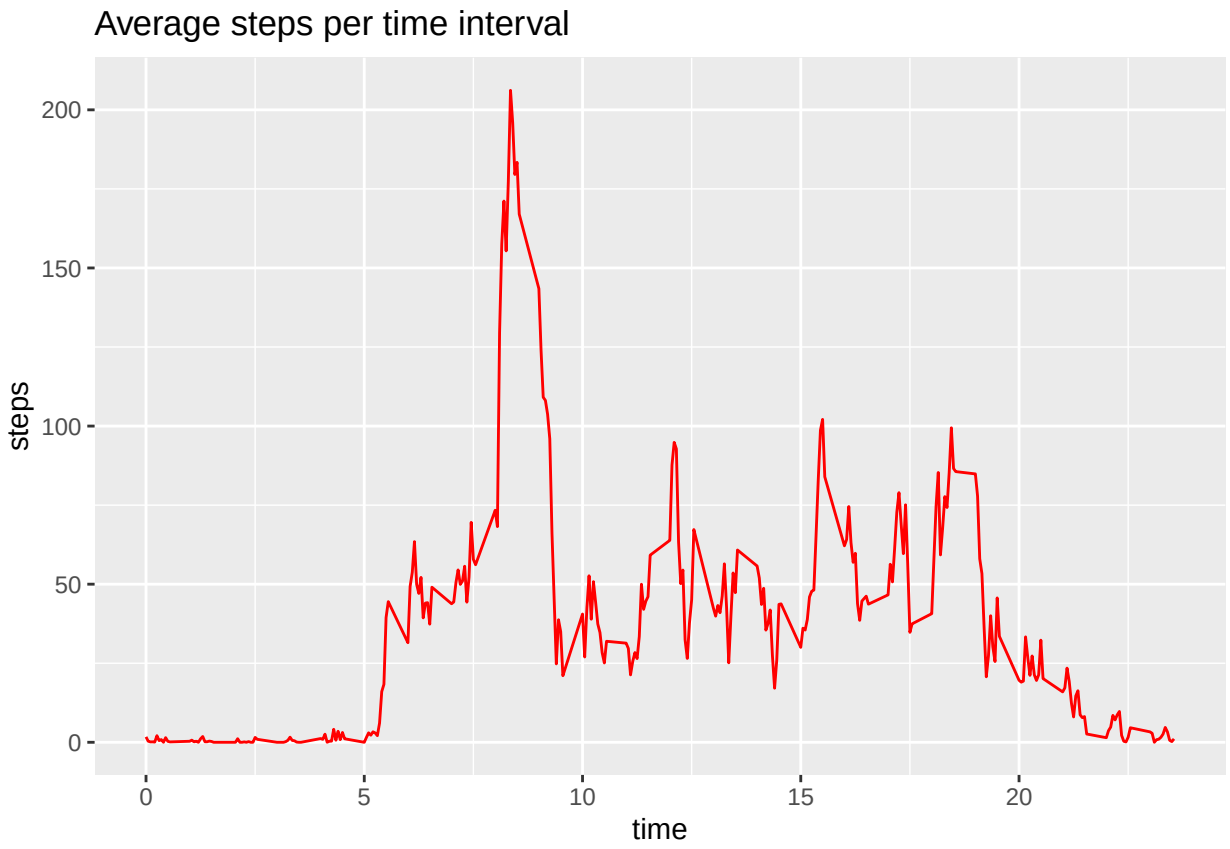
```
median(stepsperday$Steps, na.rm=TRUE)
```

```
## [1] 10765
```

2. What is the average daily activity pattern?

Time series of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
stepsperint <- aggregate(steps~interval,data=DATA,FUN=mean,na.action=na.omit)
stepsperint$time <- stepsperint$interval/100
G <- ggplot(stepsperint, aes(time, steps))
G+geom_line(col="red")+ggtitle("Average steps per time interval")
```



Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

```
library(dplyr)
M <- tbl_df(stepsperint)
```

```
## Warning: `tbl_df()` is deprecated as of dplyr 1.0.0.
## Please use `tibble::as_tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
M %>% select(time, steps) %>% filter(steps==max(M$steps))
```

```
## # A tibble: 1 x 2
##    time steps
##   <dbl> <dbl>
## 1  8.35  206.
```

3. Imputing missing values

Calculate and report the total number of missing values in the datase

```
ACT <- tbl_df(DATA)
ACT %>% filter(is.na(steps)) %>% summarize(missing_values = n())
```

```
## # A tibble: 1 x 1
##   missing_values
##            <int>
## 1           2304
```

Replace missing values

```
DATA$CompleteSteps <- ifelse(is.na(DATA$steps),
                       round(stepsperint$steps[match(DATA$interval,
                       stepsperint$interval)],0), DATA$steps)
head(DATA$CompleteSteps)
```

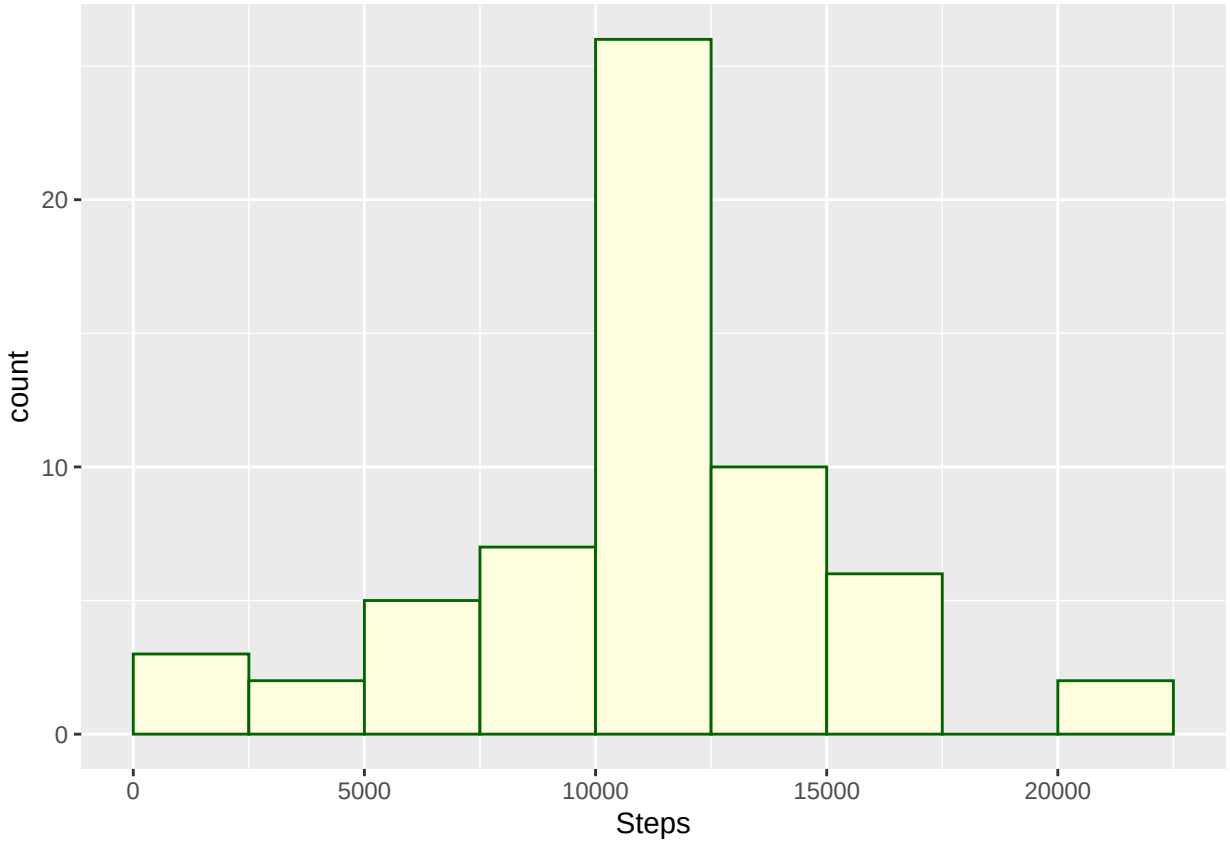
```
## [1] 2 0 0 0 0 2
```

```
## New dataset

DATAFull <- data.frame(steps=DATA$CompleteSteps,
                           interval=DATA$interval, date=DATA$date)


head(DATAFull)
```

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

Histogram of the total number of steps taken each day with missing data filled in

```
Full <- aggregate(DATAFull$steps, list(DATAFull$date), FUN=sum)
colnames(Full) <- c("Date", "Steps")

H2 <- ggplot(Full, aes(Steps))
H2+geom_histogram(boundary=0, binwidth=2500, col="darkgreen", fill="lightyellow")
```



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

```
mean(Full$Steps)
```

```
## [1] 10765.64
```

```
median(Full$Steps)
```

```
## [1] 10762
```

```
# Both decreased slightly
```

4. Are there differences in activity patterns between weekdays and weekends?

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

```
DATAFull$RealDate <- as.Date(DATAFull$date, format = "%Y-%m-%d")
DATAFull$weekday <- weekdays(DATAFull$RealDate)



for(i in 1:length(DATAFull$date)){
  if(weekdays(as.Date(DATAFull$date[i]))=="Sábado"|weekdays(as.Date(DATAFull$date[i]))=="Domingo"){
    DATAFull$day[i]="weekend"
  }
  else{
    DATAFull$day[i]="weekday"
  }
}

head(DATAFull, n=10)
```

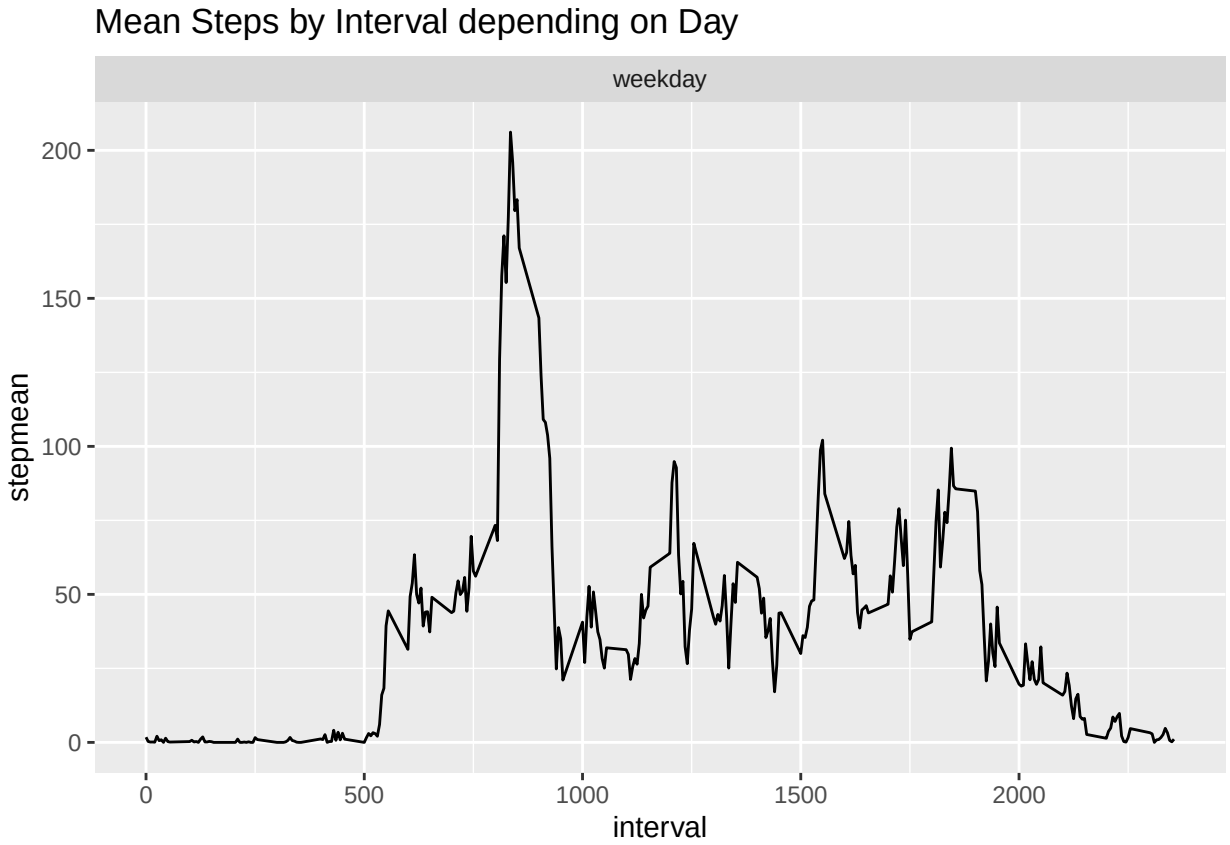
```
##    steps interval       date   RealDate weekday     day
## 1      2        0 2012-10-01 2012-10-01   lunes weekday
## 2      0        5 2012-10-01 2012-10-01   lunes weekday
## 3      0       10 2012-10-01 2012-10-01   lunes weekday
## 4      0       15 2012-10-01 2012-10-01   lunes weekday
## 5      0       20 2012-10-01 2012-10-01   lunes weekday
## 6      2       25 2012-10-01 2012-10-01   lunes weekday
## 7      1       30 2012-10-01 2012-10-01   lunes weekday
## 8      1       35 2012-10-01 2012-10-01   lunes weekday
## 9      0       40 2012-10-01 2012-10-01   lunes weekday
## 10     1       45 2012-10-01 2012-10-01   lunes weekday
```

Two time series plot of the 5-minute interval (x) and the average number of steps taken averaged across weekday days or weekend days (y).

```
weekdata<-DATAFull %>% group_by(day,interval)%>% summarise(stepmean=mean(steps))
```

```
## `summarise()` regrouping output by 'day' (override with `.groups` argument)
```

```
ggplot(weekdata,aes (interval, stepmean)) + geom_line() +facet_wrap(day~.,nrow=2,ncol=1)+ggtitle("Mean S
```


Mean Steps by Interval depending on Day
weekday
stepmean
200
150
100
50
0
0
500
1000
1500
2000
interval