Kutessa Garnett Reproducible Research: Peer Assessment 1

2022-11-03

## R Markdown

#Load the data

df<- read.csv("activity.csv", na.strings="NA")  
head(df)

## 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

#Process/transform the data  
df$date <- as.Date(df$date)  
df\_stepping<-subset(df, !is.na(df$steps))  
head(df$date)

## [1] "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01"  
## [6] "2012-10-01"

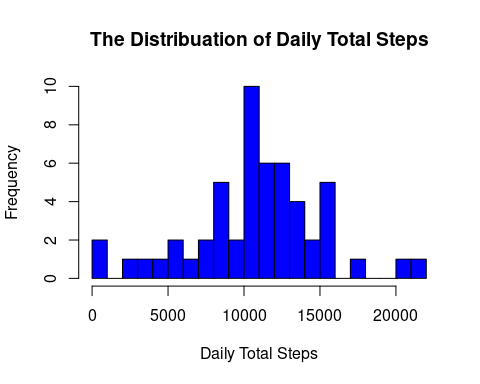
head(df\_stepping)

## steps date interval  
## 289 0 2012-10-02 0  
## 290 0 2012-10-02 5  
## 291 0 2012-10-02 10  
## 292 0 2012-10-02 15  
## 293 0 2012-10-02 20  
## 294 0 2012-10-02 25

#Number of steps taken per day  
stepsday <- tapply(df\_stepping$steps, df\_stepping$date, sum, na.rm=TRUE, simplify =T)  
stepsday

## 2012-10-02 2012-10-03 2012-10-04 2012-10-05 2012-10-06 2012-10-07 2012-10-09   
## 126 11352 12116 13294 15420 11015 12811   
## 2012-10-10 2012-10-11 2012-10-12 2012-10-13 2012-10-14 2012-10-15 2012-10-16   
## 9900 10304 17382 12426 15098 10139 15084   
## 2012-10-17 2012-10-18 2012-10-19 2012-10-20 2012-10-21 2012-10-22 2012-10-23   
## 13452 10056 11829 10395 8821 13460 8918   
## 2012-10-24 2012-10-25 2012-10-26 2012-10-27 2012-10-28 2012-10-29 2012-10-30   
## 8355 2492 6778 10119 11458 5018 9819   
## 2012-10-31 2012-11-02 2012-11-03 2012-11-05 2012-11-06 2012-11-07 2012-11-08   
## 15414 10600 10571 10439 8334 12883 3219   
## 2012-11-11 2012-11-12 2012-11-13 2012-11-15 2012-11-16 2012-11-17 2012-11-18   
## 12608 10765 7336 41 5441 14339 15110   
## 2012-11-19 2012-11-20 2012-11-21 2012-11-22 2012-11-23 2012-11-24 2012-11-25   
## 8841 4472 12787 20427 21194 14478 11834   
## 2012-11-26 2012-11-27 2012-11-28 2012-11-29   
## 11162 13646 10183 7047

#Histogram of total number of steps taken per day  
hist(x=stepsday, col="blue", breaks =20, xlab="Daily Total Steps", ylab="Frequency", main="The Distribuation of Daily Total Steps")



#Mean of steps  
mean(stepsday)

## [1] 10766.19

#Median of steps  
median(stepsday)

## [1] 10765

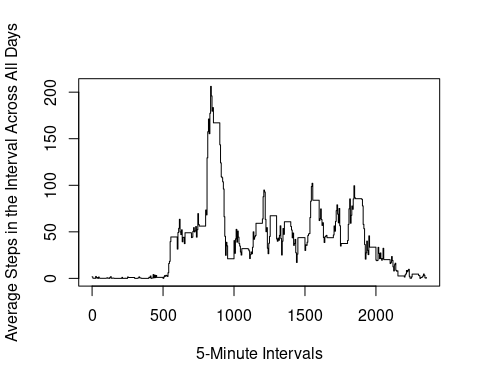
#Time series plot of 5 minute interval  
int\_avg<-tapply(df\_stepping$steps, df\_stepping$interval, mean, na.rm=TRUE, simplify=T)  
df\_ia<- data.frame(interval=as.integer(names(int\_avg)), avg=int\_avg)  
head(int\_avg)

## 0 5 10 15 20 25   
## 1.7169811 0.3396226 0.1320755 0.1509434 0.0754717 2.0943396

head(df\_ia)

## interval avg  
## 0 0 1.7169811  
## 5 5 0.3396226  
## 10 10 0.1320755  
## 15 15 0.1509434  
## 20 20 0.0754717  
## 25 25 2.0943396

with(df\_ia, plot(interval, avg, type="s", xlab="5-Minute Intervals", ylab="Average Steps in the Interval Across All Days"))



#Which 5 minute interval contains the maximum number of step  
max\_steps5 <- max(df\_ia$avg)  
df\_ia[df\_ia$avg==max\_steps5,]

## interval avg  
## 835 835 206.1698

head(max\_steps5)

## [1] 206.1698

#Numberof missing values  
sum(is.na(df$steps))

## [1] 2304

#Filling in missing values  
df\_addinvalue<-df  
ndx<-is.na(df\_addinvalue$steps)  
int\_avg<-tapply(df\_stepping$steps, df\_stepping$interval, mean, na.rm=TRUE, simplify=T)  
df\_addinvalue$steps[ndx]<-int\_avg[as.character(df\_addinvalue$interval[ndx])]  
head(df\_addinvalue)

## steps date interval  
## 1 1.7169811 2012-10-01 0  
## 2 0.3396226 2012-10-01 5  
## 3 0.1320755 2012-10-01 10  
## 4 0.1509434 2012-10-01 15  
## 5 0.0754717 2012-10-01 20  
## 6 2.0943396 2012-10-01 25

#Histogram with missing data numbers  
stepsdaymiss <- tapply(df\_addinvalue$steps, df\_addinvalue$date, sum, na.rm=TRUE, simplify =T)  
stepsdaymiss

## 2012-10-01 2012-10-02 2012-10-03 2012-10-04 2012-10-05 2012-10-06 2012-10-07   
## 10766.19 126.00 11352.00 12116.00 13294.00 15420.00 11015.00   
## 2012-10-08 2012-10-09 2012-10-10 2012-10-11 2012-10-12 2012-10-13 2012-10-14   
## 10766.19 12811.00 9900.00 10304.00 17382.00 12426.00 15098.00   
## 2012-10-15 2012-10-16 2012-10-17 2012-10-18 2012-10-19 2012-10-20 2012-10-21   
## 10139.00 15084.00 13452.00 10056.00 11829.00 10395.00 8821.00   
## 2012-10-22 2012-10-23 2012-10-24 2012-10-25 2012-10-26 2012-10-27 2012-10-28   
## 13460.00 8918.00 8355.00 2492.00 6778.00 10119.00 11458.00   
## 2012-10-29 2012-10-30 2012-10-31 2012-11-01 2012-11-02 2012-11-03 2012-11-04   
## 5018.00 9819.00 15414.00 10766.19 10600.00 10571.00 10766.19   
## 2012-11-05 2012-11-06 2012-11-07 2012-11-08 2012-11-09 2012-11-10 2012-11-11   
## 10439.00 8334.00 12883.00 3219.00 10766.19 10766.19 12608.00   
## 2012-11-12 2012-11-13 2012-11-14 2012-11-15 2012-11-16 2012-11-17 2012-11-18   
## 10765.00 7336.00 10766.19 41.00 5441.00 14339.00 15110.00   
## 2012-11-19 2012-11-20 2012-11-21 2012-11-22 2012-11-23 2012-11-24 2012-11-25   
## 8841.00 4472.00 12787.00 20427.00 21194.00 14478.00 11834.00   
## 2012-11-26 2012-11-27 2012-11-28 2012-11-29 2012-11-30   
## 11162.00 13646.00 10183.00 7047.00 10766.19

hist(x=stepsdaymiss, col="blue", breaks =20, xlab="Daily Total Steps", ylab="Frequency", main="The Distribuation of Daily Total Steps")  
  
  
  
#Mean of steps  
mean(stepsdaymiss)

## [1] 10766.19

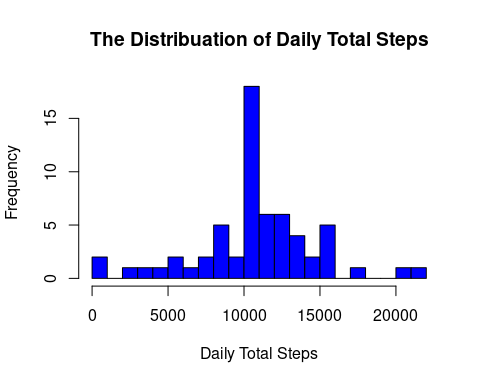
#Median of steps  
median(stepsdaymiss)

## [1] 10766.19

#The mean is the same and the median changed to to be closer to the mean  
  
#Are there differences in activity patterns between weekdays and weekends?  
  
isitweekday<-function(d){wd<-weekdays(d)   
ifelse (wd=="Saturday" | wd=="Sunday", "weekend", "weekday")}  
  
wx<-sapply(df\_addinvalue$date, isitweekday)  
df\_addinvalue$wk<-as.factor(wx)  
head(df\_addinvalue)

## steps date interval wk  
## 1 1.7169811 2012-10-01 0 weekday  
## 2 0.3396226 2012-10-01 5 weekday  
## 3 0.1320755 2012-10-01 10 weekday  
## 4 0.1509434 2012-10-01 15 weekday  
## 5 0.0754717 2012-10-01 20 weekday  
## 6 2.0943396 2012-10-01 25 weekday

#Panel plot with time series plot of the 5 minute interval  
  
wk\_df <- aggregate(steps ~ wk+interval, data=df\_addinvalue, FUN=mean)  
  
library(lattice)



xyplot(steps ~ interval | factor(wk),  
 layout = c(1, 2),  
 xlab="Interval",  
 ylab="Number of steps",  
 type="l",  
 lty=1,  
 data=wk\_df)

