2020_0906_Cour_Repro_Res_Mov_Proj

John Nguyen

9/6/2020

```
data<-read.csv("activity.csv")
head(data)
```

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

Loading and preprocessing the data

This will include code cleaning

```
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
```

```
data<-read.csv("activity.csv")
data$date <- as.Date(data$date, format = "%m/%d/%Y")
```

What is mean total number of steps taken per day?

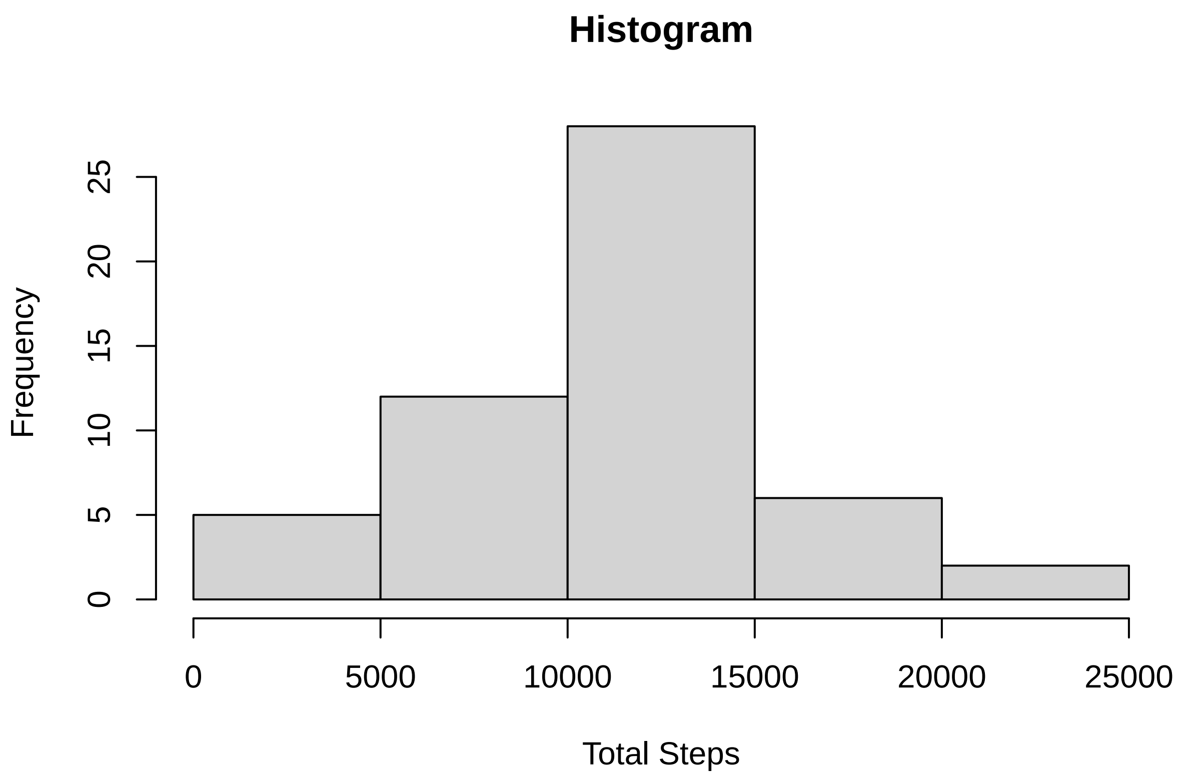
```
##1) Calculate the total number of steps taken per day
datasumdate<-data%>%
    group_by(date)%>%
    summarise(sumdate<-sum(steps))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
datasumdate<-data.frame(datasumdate)
head(datasumdate)
```

```
##         date sumdate....sum.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
```

```
##2) If you do not understand the difference between a histogram and a barplot, research the difference
hist(datasumdate$sumdate....sum.steps., main='Histogram', xlab='Total Steps')
```



```
##3) Calculate and report the mean and median of the total number of steps taken per day

mean(datasumdate$sumdate....sum.steps., na.rm = TRUE) ##Answer
```

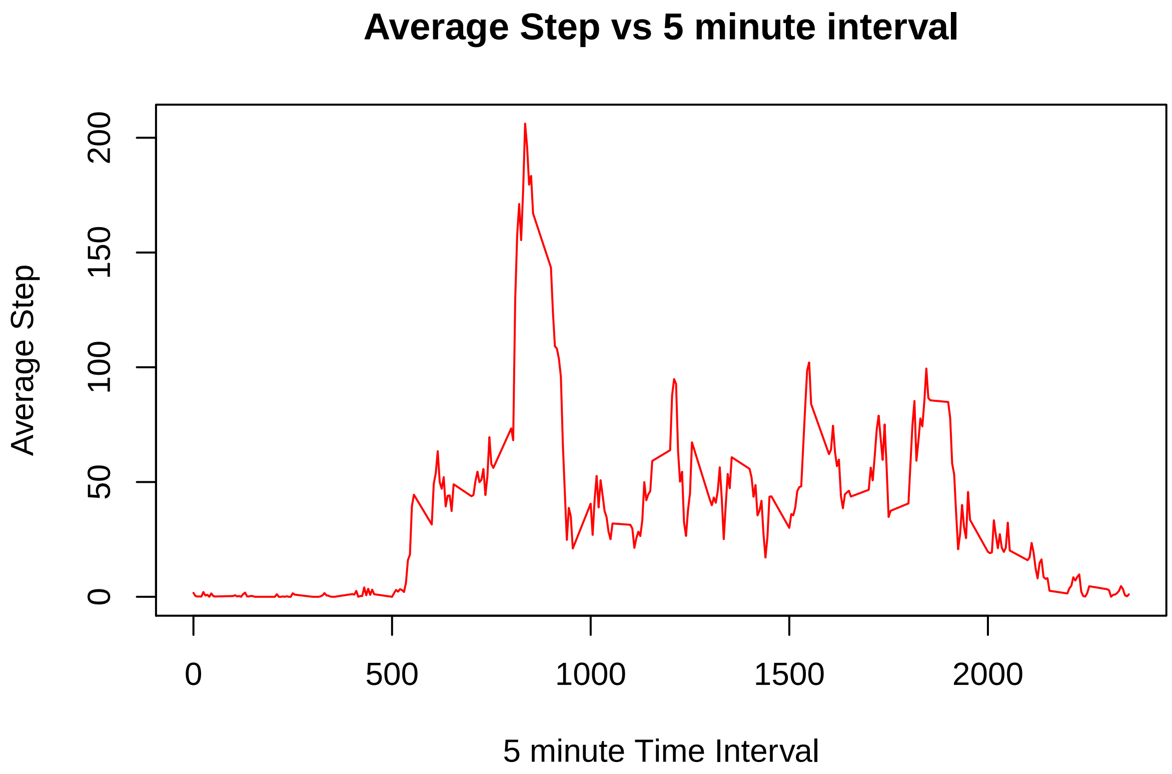
```
## [1] 10766.19
```

```
median(datasumdate$sumdate....sum.steps., na.rm=TRUE) ##Answer
```

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

What is the average daily activity pattern?

```
##Time series plot of the average number of steps taken
avg_step<-tapply(data$steps,data$interval,FUN = mean,na.rm=TRUE)
avg_step<-data.frame(avg_step)
df2 <- cbind(interval = rownames(avg_step), avg_step)
rownames(df2) <- 1:nrow(df2)
plot(df2$interval,df2$avg_step, type='l',xlab='5 minute Time Interval',ylab='Average Step', col='red',
     main='Average Step vs 5 minute interval')
```



```
## Contains the maximum number of steps
highest_interval<-subset(df2, avg_step==max(avg_step))
highest_interval
```

```
##     interval avg_step
## 104      835 206.1698
```

Imputing missing values

```
##1) Calculate and report the total number of missing values in the dataset.
sum(is.na(data))
```

```
## [1] 2304
```

```
##2) Devise a strategy for filling in all of the missing values in the dataset. I chose the median for
med_step<-aggregate(steps~interval, data=data, median, na.rm=TRUE)
b<-numeric()
for (i in 1:nrow(data))
{
    if(is.na(data[i,]$steps)){##if steps in data is na
        a<-subset(med_step,interval==data[i,]$interval)$steps
    }
    else{
        a<-data[i,]$steps
    }
    b<-c(b,a) ##concat and keep every value of a stored in b
}
b<-data.frame(b)

##3) Create a new dataset that is equal to the original dataset but with the missing data filled in.

new_data<-c(data,b)

new_data<-data.frame(new_data)
new_data2<-new_data[,2:4]
names(new_data2)[names(new_data2)=='b']<-'steps'
str(new_data2)
```

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

```
##4) Make a histogram of the total number of steps taken each day and Calculate and report the mean and

datasum2<-new_data2%>%
    group_by(date)%>%
    summarise(sumdate<-sum(steps))
```

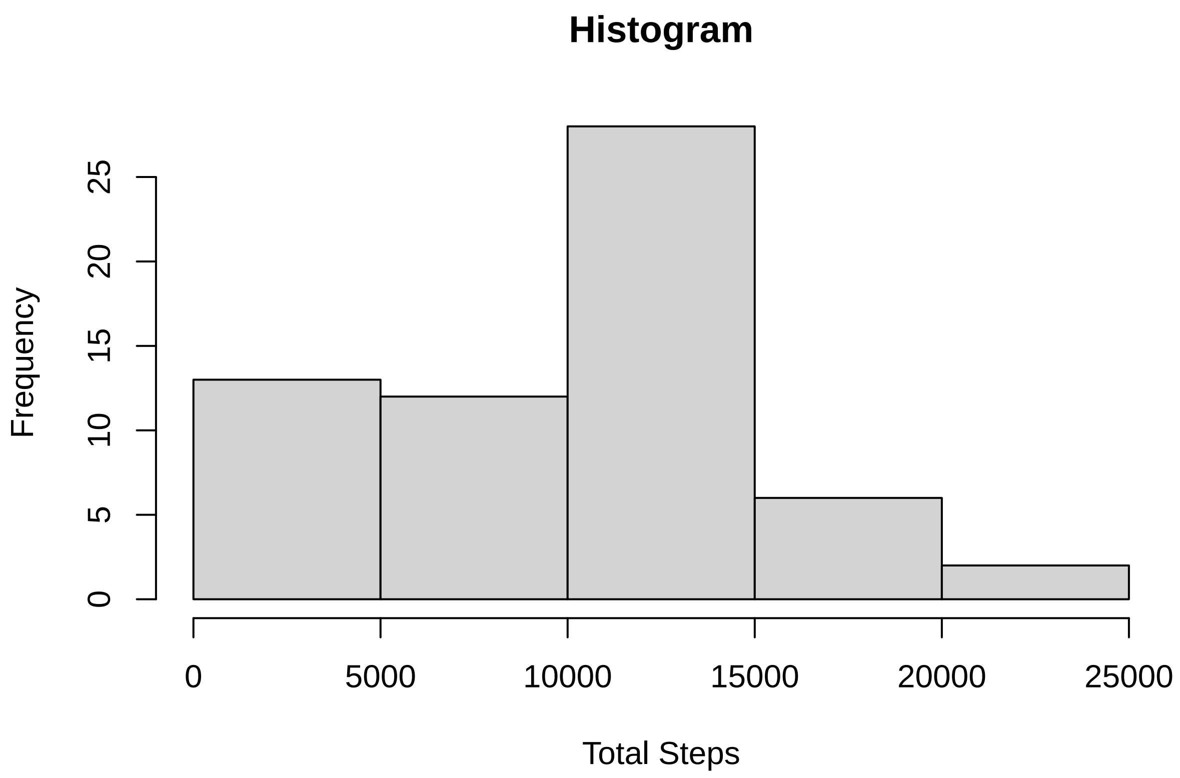
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
head(datasum2)
```

```
## # A tibble: 6 x 2
##   date       `sumdate <- sum(steps)`
##   <date>                       <dbl>
## 1 2012-10-01                    1141
## 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
```

```
datasum2<-data.frame(datasum2)

hist(datasum2$sumdate....sum.steps., main='Histogram', xlab='Total Steps')
```



```r
mean(datasum2$sumdate....sum.steps.) ##mean
```

```
## [1] 9503.869
```

```r
median(datasum2$sumdate....sum.steps.)##median
```

```
## [1] 10395
```

```r
##Comment: We find that both mean and median drops when we replace missing values in steps with the med
```

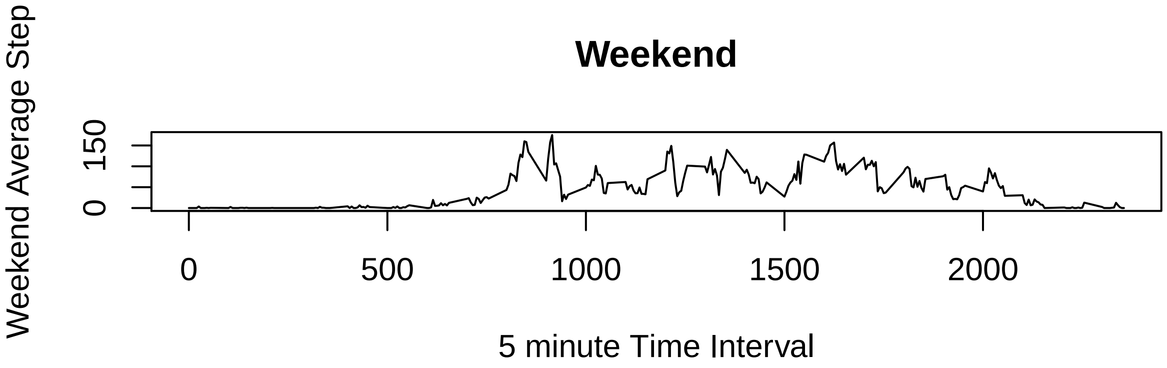
Are there differences in activity patterns between weekdays and weekends?

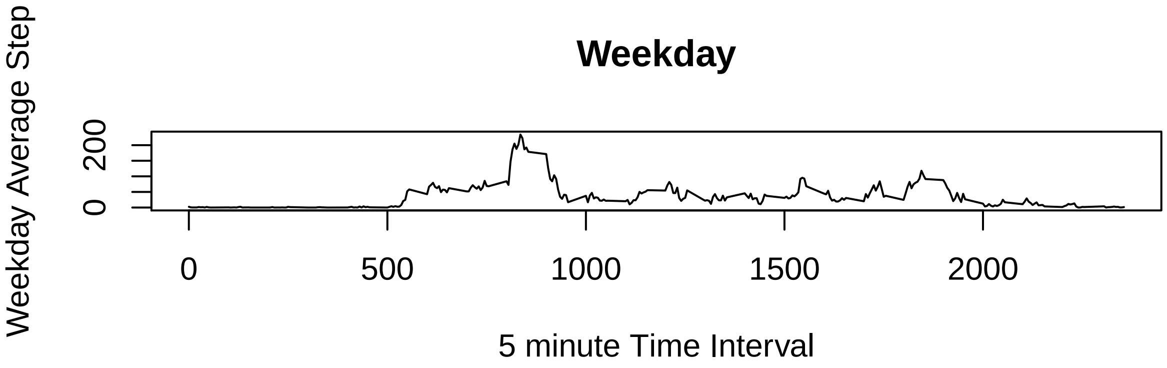
```r
##1)Create a new factor variable in the dataset with two levels - "weekday" and "weekend" indicating wh
data$days<-weekdays(data$date)

dataweekend<-subset(data,days=='Saturday'| days=='Sunday')
dataweekend$days<-as.factor(dataweekend$days)
avg_weekend<-aggregate(steps~interval, data=dataweekend, mean)

dataweek<-subset(data, days!='Saturday' & days!='Sunday')
dataweek$days<-as.factor(dataweek$days)
avg_week<-aggregate(steps~interval, data=dataweek, mean)


par(mfrow=c(2,1))
plot(avg_weekend$interval,avg_weekend$steps, type='l',xlab='5 minute Time Interval',ylab='Weekend Averag
plot(avg_week$interval,avg_week$steps, type='l',xlab='5 minute Time Interval',ylab='Weekday Average Ste
```


Weekend
Weekend Average Step
0
150
0
500
1000
1500
2000
5 minute Time Interval


Weekday
Weekday Average Step
0
200
0
500
1000
1500
2000
5 minute Time Interval