ASSIGNMENT 1_RATHsid

This is the report for the Assignment 1 of the Reproducible Research Course, as part of the Data Science Specialization offered by John Hopkin’s Universty.The assignment is answered part after part in chronological order.

Loading and preprocessing the data

Part1 The .csv file is read and assigned to the variable- data. Then the basic structure of the data is examined.

```
data<-read.csv("activity.csv",header=TRUE)
dim(data)
```

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

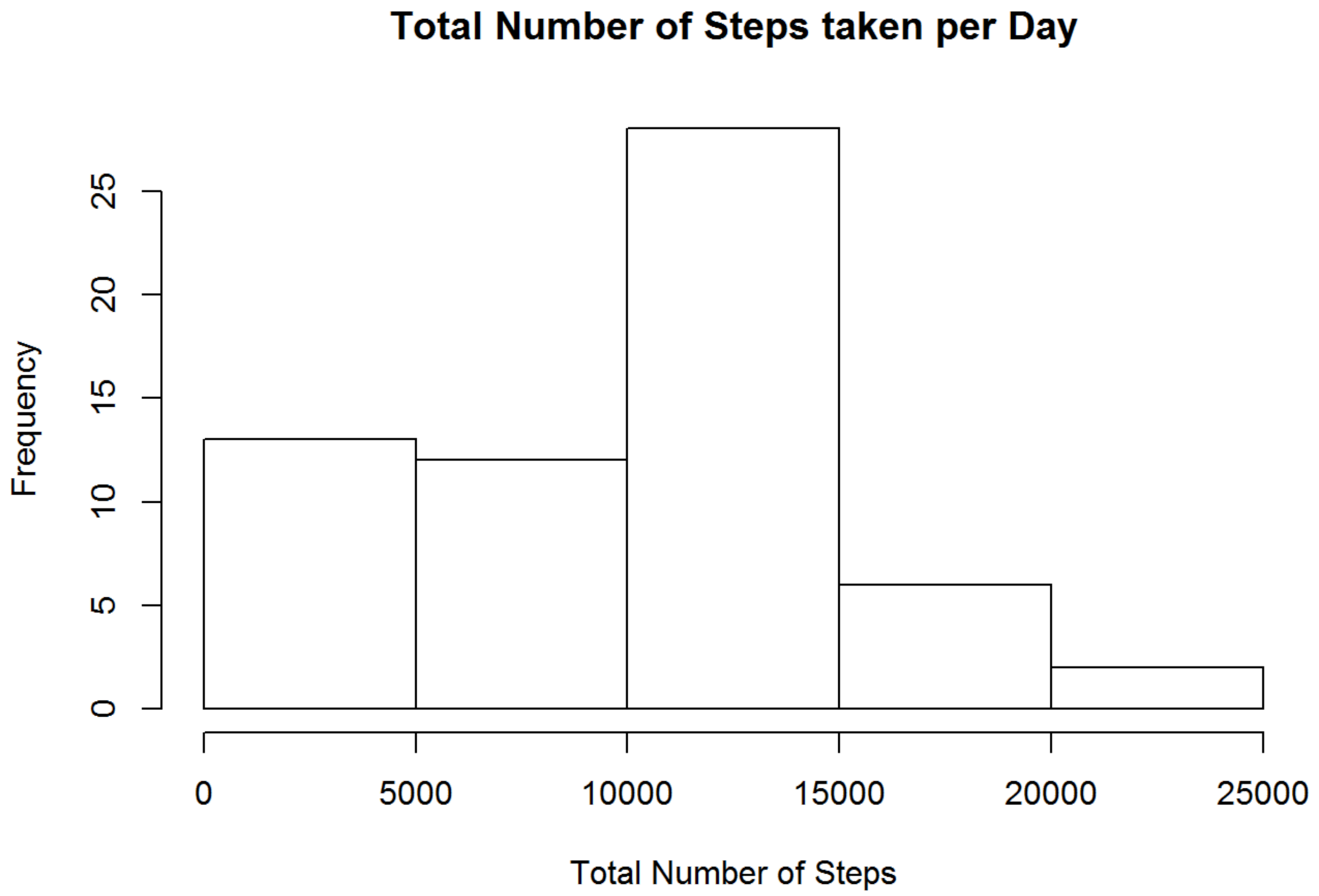
```
head(data)
```

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

What is mean total number of steps taken per day?

Part 1 To make the histogram of the- Total number of steps each day, first we remove the NAs from the dataset. Then we split the data datewise and take the sum of steps for each of the dates. After that the same is converted into a histogram which shows the frequency distribution of Total number of steps on the various dates.

```
data1<-data[!is.na(data[,1]),]
total1<-split(data1$steps,data1$date)
totalsteps1<-lapply(total1,sum)
hist(as.numeric(totalsteps1), xlab="Total Number of Steps",main = "Total Number of Steps taken per
 Day")
```



Part 2 The next step is to calculate the mean and median of the above data set i.e the Total Number of Steps taken per Day. For that, we will not consider the observations that are marked NA, the **mean and median are calculated only on the valid observations**.

```
totalsteps1<-as.data.frame(totalsteps1)
totalsteps1<-t(totalsteps1)
mean(totalsteps1[!is.na(totalsteps1[,1]),1])
```

```
## [1] 9354
```

```
median(totalsteps1[!is.na(totalsteps1[,1]),1])
```

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

So the mean and median of the above data are **9354 and 10395**.

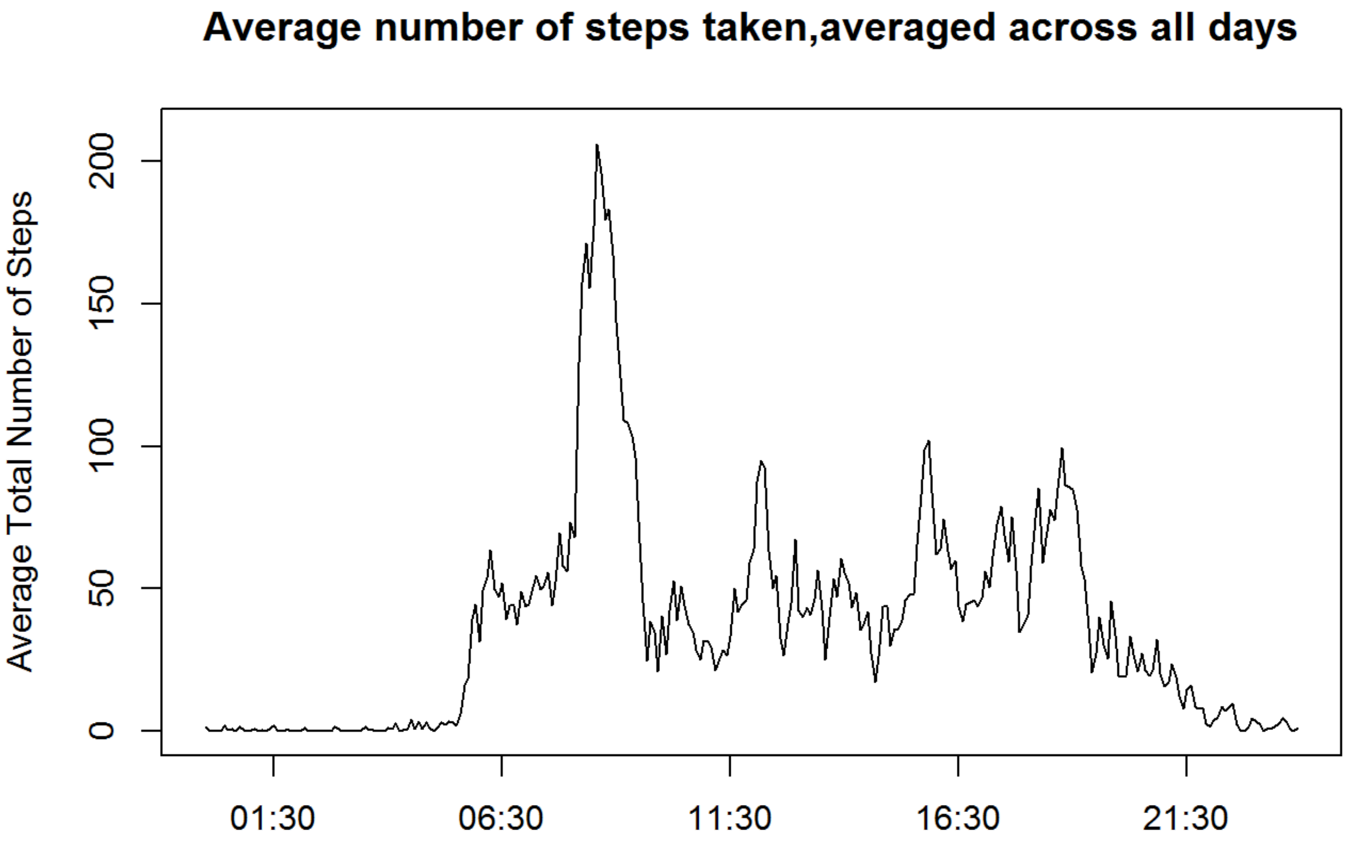
What is the average daily activity pattern?

Part 1 The next task is to make a time series plot of average number of steps taken, averaged across all days. For that we split the data set based on the interval and take the average.

```
total2<-split(data1$steps,data1$interval)
totalsteps2<-lapply(total2,mean)
interval<-names(totalsteps2)
totalsteps2<-as.data.frame(totalsteps2)
totalsteps2<-t(totalsteps2)
totalsteps2<-cbind(interval,totalsteps2[,1])
totalsteps2<-as.data.frame(totalsteps2)
```

Then we will convert the intrerval observations into class - time so that it could be converted into a time series plot. Then next we plot a time series with the average number of steps taken in a particular time interval, averaged across all days.

```
i<-1
datetime1<-vector()
totalsteps2[,1]<-as.character(totalsteps2[,1])
while(i<=288){

h<-as.character(as.numeric(totalsteps2[i,1])%/%100)
m<-as.character(as.numeric(totalsteps2[i,1])%%100)
s<-"00"
time1<-paste(h,m,s,sep=":")
totalsteps2[i,1]<-time1

i<-i+1
}
datetime<-strptime(totalsteps2[,1], "%H:%M:%S")
steps<-as.numeric(as.character(totalsteps2[,2]))
plot(datetime,steps, type = "l", ylim = c(0,210), xlab=" ", ylab="Average Total Number of Steps",main="Average number of steps taken,averaged across all days")
```



Part 2 Next we find the 5-minute interval, on an average across all the days in the dataset, that contains the maximum number of steps.

```
msteps<-max(as.numeric(as.character(totalsteps2[,2])))
totalsteps2[as.numeric(as.character(totalsteps2[,2]))==msteps,]
```

```
##      interval              V2
## X835  8:35:00 206.169811320755
```

Hence the interval that contains the maximum number of steps is **8:35am-8:40am**, the average steps in this interval being **206.17**.

Imputing missing values

Part1 The next task is to fnd the number of missing values or the NAs.

```
 sum(is.na(data[,1]))
```

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

So the number of missing values are **2304**.

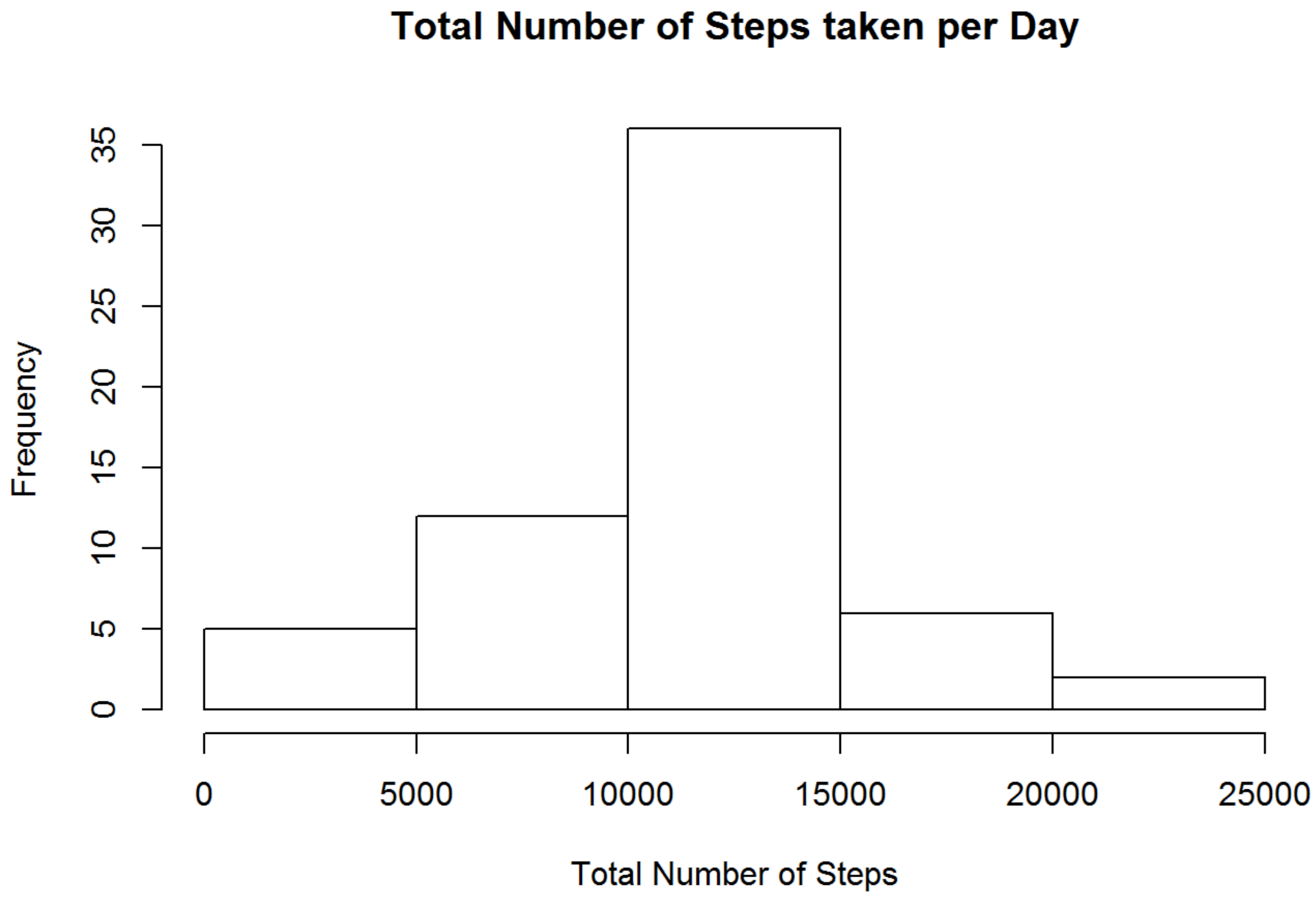
Part2 & Part3 Next we devise a startegy to fill in the mssing NA values. For that the most **appropriate strategy** would be probably to assign them the average values of steps taken at the respective intervals averaged across all days. So we create a dataset- data2 which we initially assign the data we read, then we assign the NAs with values that are the average of the number of steps taken at a specific interval averaged across all days.

```
data2<-data


total3<-split(data1$steps,data1$interval)
totalsteps3<-lapply(total3,mean)
interval<-names(totalsteps3)
totalsteps3<-as.data.frame(totalsteps3)
totalsteps3<-t(totalsteps3)
totalsteps3<-cbind(interval,totalsteps3[,1])
totalsteps3<-as.data.frame(totalsteps3)


j<-1
while(j<=17568){


if(is.na(data2[j,1])){


  r<-data2[j,3]


  data2[j,1]<-as.numeric(as.character(totalsteps3[totalsteps3[,1]==r,2]))
  j<-j+1
}else if(!is.na(data2[j,1])){
  j<-j+1
}else{}
}


head(data2)
```

```
##     steps       date interval
## 1 1.71698 01/10/2012        0
## 2 0.33962 01/10/2012        5
## 3 0.13208 01/10/2012       10
## 4 0.15094 01/10/2012       15
## 5 0.07547 01/10/2012       20
## 6 2.09434 01/10/2012       25
```

Part4 Next we plot the histogram with the newly created data set.

```
total4<-split(data2$steps,data2$date)
totalsteps4<-lapply(total4,sum)
hist(as.numeric(totalsteps4), xlab="Total Number of Steps",main = "Total Number of Steps taken per
 Day")
```



Then we calculate the mean and median total number of steps taken per day, with the new data set.

```
totalsteps4<-as.data.frame(totalsteps4)
totalsteps4<-t(totalsteps4)
mean(as.numeric(as.character(totalsteps4[!is.na(totalsteps4[,1]),1])))
```

```
## [1] 10766
```

```
median(as.numeric(as.character(totalsteps4[!is.na(totalsteps4[,1]),1])))
```

```
## [1] 10766
```

Yes the mean and median differ from the first part of the assignment. The values increase from the previous values and at the same time they converge towards each other as well, so that they are almost equal the value of the mean and median being very close to **10766**.

Are there differences in activity patterns between weekdays and weekends?

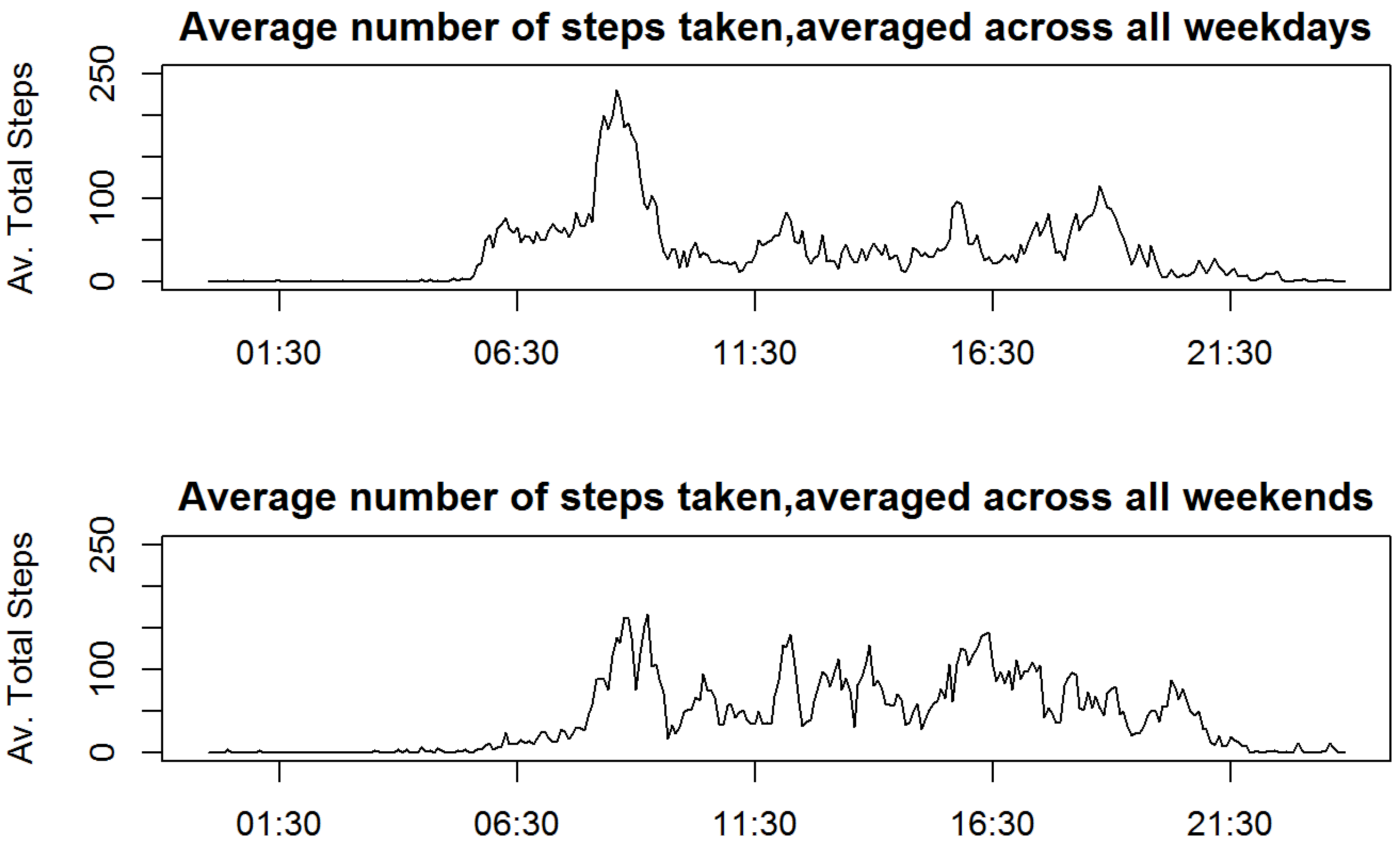
Part1 *For this part of the assignment we assume both Saturday and Sunday as weekend days and rest of the days as weekdays.* We will first assign the dates with the respective days with the help of the **weekdays()** and then we assign Saturday and Sunday as **Weekend** and rest of the days as **Weekday**

```
data3<-data2
k<-1
datetime3<-vector()
ori_int<-data3[,3]
data3<-cbind(data3,ori_int)

data3[,3]<-as.character(data3[,3])
while(k<=17568){

  h1<-as.character(as.numeric(as.character(data3[k,3]))%/%100)
  m1<-as.character(as.numeric(as.character(data3[k,3]))%%100)
  s1<-"00"
  d1<-as.character(data3[k,2])

  time1<-paste(h1,m1,s1,sep=":")
  time1<-paste(d1,time1)
  data3[k,3]<-time1

   k<-k+1
}

datetime3<-strptime(data3[,3], "%d/%m/%Y %H:%M:%S")
day<-weekdays(datetime3)
data3<-cbind(data3,day)
data3[,5]<-as.character(data3[,5])
l<-1
data3[,5]<-as.character(data3[,5])




data3[((data3[,5]=="Sunday") | (data3[,5]=="Saturday")),5]<-"Weekend"
data3[((data3[,5]=="Monday") | (data3[,5]=="Tuesday") | (data3[,5]=="Wednesday") | (data3[,5]=="Thursday") | (data3[,5]=="Friday")),5]<-"Weekday"

data3[1437:1447,-3]
```

```
##      steps       date ori_int     day
## 1437     0 05/10/2012    2340 Weekday
## 1438     0 05/10/2012    2345 Weekday
## 1439     0 05/10/2012    2350 Weekday
## 1440     0 05/10/2012    2355 Weekday
## 1441     0 06/10/2012       0 Weekend
## 1442     0 06/10/2012       5 Weekend
## 1443     0 06/10/2012      10 Weekend
## 1444     0 06/10/2012      15 Weekend
## 1445     0 06/10/2012      20 Weekend
## 1446     0 06/10/2012      25 Weekend
## 1447     0 06/10/2012      30 Weekend
```

Part2 Next we plot a time series plot for Average number of steps taken, averaged across all the days separately for Weekdays and Weekends.

```
data4<-data3

total5<-split(data4,data4$day)

weekday<-total5[[1]][,-5][,-3][,-2]
weekend<-total5[[2]][,-5][,-3][,-2]

wd<-split(weekday$steps,weekday$ori_int)
wd1<-as.data.frame(lapply(wd,mean))
wd1<-t(wd1)

we<-split(weekend$steps,weekend$ori_int)
we1<-as.data.frame(lapply(we,mean))
we1<-t(we1)

par(mfrow = c(2, 1), mar = c(4, 4, 2, 1), oma = c(0, 0, 2, 0))
plot(datetime,wd1[,1], type = "l", ylim = c(0,250), xlab=" ", ylab="Av. Total Steps",main="Average
 number of steps taken,averaged across all weekdays")
plot(datetime,we1[,1], type = "l", ylim = c(0,250), xlab=" ", ylab="Av. Total Steps",main="Average
 number of steps taken,averaged across all weekends")
```



From the above plots we can compare and observe that the number of steps is comparatively more distributed uniformly during the day time and evening on weekends than weekdays. And on weekdays we can observe a peak durng the early morning hours probably durng the time when most people would be getting ready and starting for work.

Thats all for this assignment. Thank You!!!