Activity Monitoring Data

Piyush

Saturday, November 15, 2014

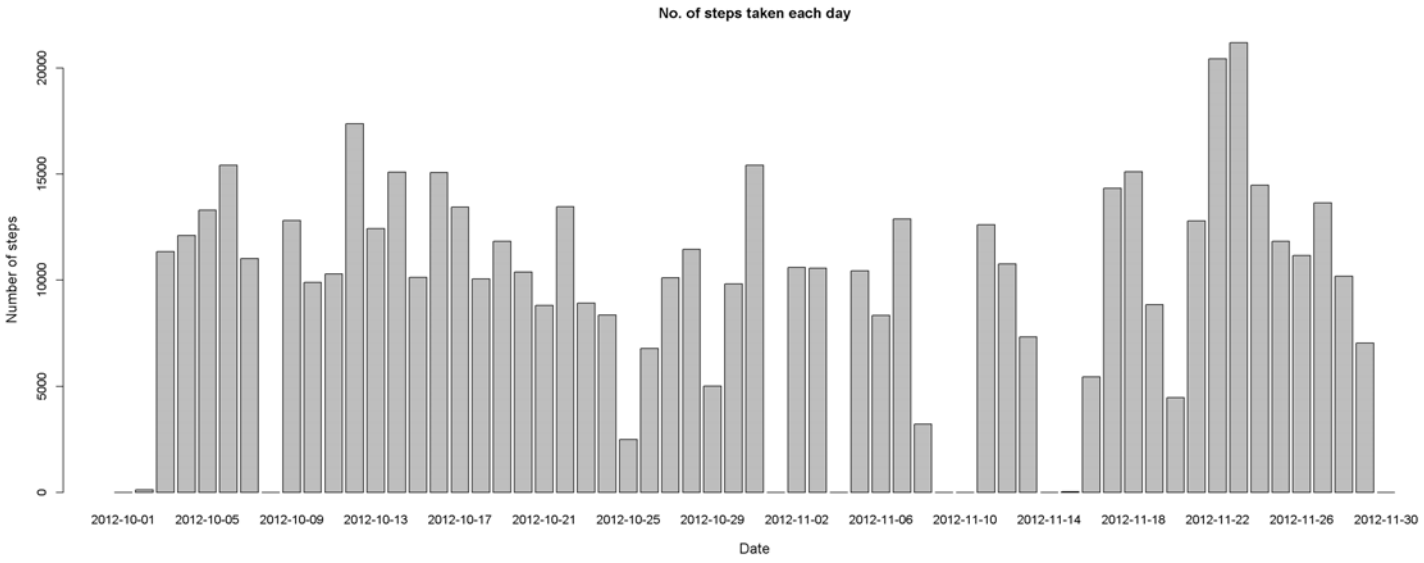
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

Loading the csv file

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

Processing the data by splitting the steps according to days and creating a date variable form the dates in the csv file. A histogram is then plotted for total number of steps taken per day

```
daily_steps<-sapply(split(file[,1],file[,2]),sum,na.rm=TRUE)
list_dates<-as.Date(names(daily_steps),"%Y-%m-%d")
barplot(daily_steps,xlab="Date", ylab="Number of steps", main="No. of steps taken each day", na
mes.arg=list_dates, cex.lab=1.2)
```

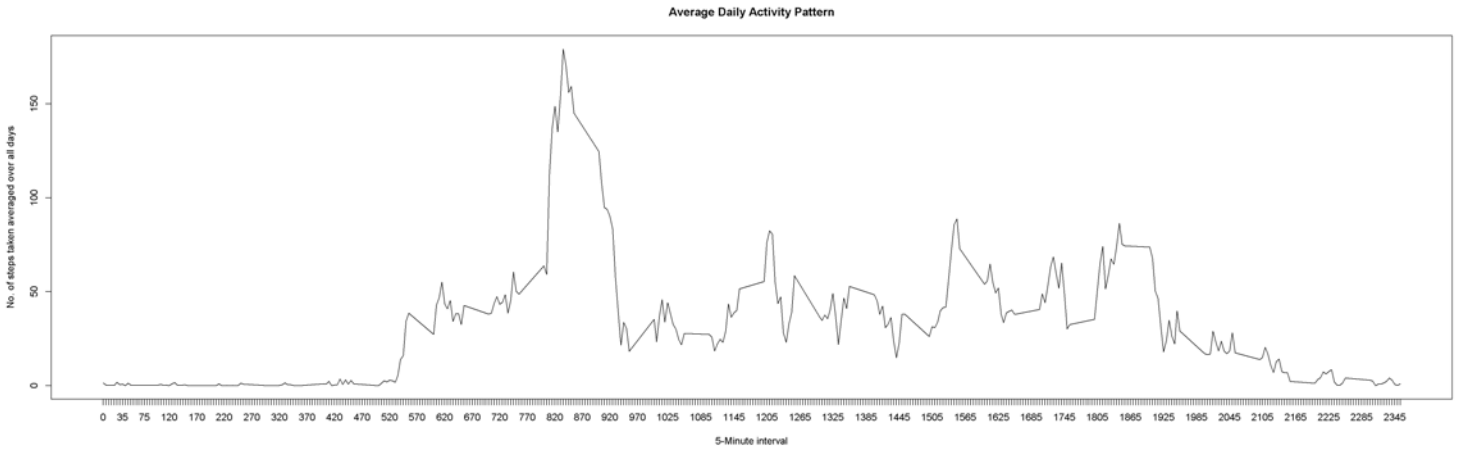



```
m1<-mean(daily_steps)
m2<-median(daily_steps)
```

The mean and median for the total number of steps per day is 9354.2295 and 10395

Data is now split according to the five minute intervals for all days and the steps taken in that interval, for all the days into consideration, is summed up

```
daily_steps2<-sapply(split(file[,1],file[,3]),sum,na.rm=TRUE)
daily_steps2<-daily_steps2/length(levels(file[,2]))
plot(x=names(daily_steps2), y=daily_steps2, type="l", main="Average Daily Activity Pattern", xl
ab="5-Minute interval", ylab="No. of steps taken averaged over all days", xaxp=c(0,2355,471))
```

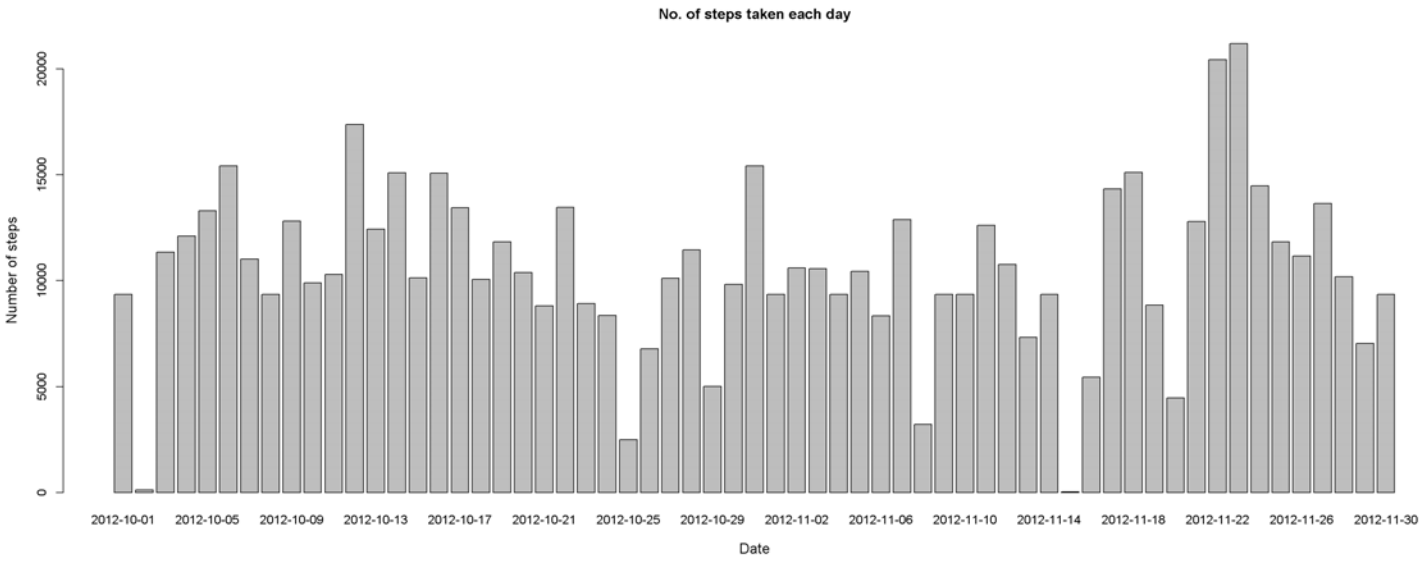



```
max_steps<-max(daily_steps2)
max_steps_int<-names(which(daily_steps2==max_steps))
```

The maximum number of steps taken in any 5 min interval is 179.1311 and the 5-min interval for which it occurs is 835

Replacing the missing value for a row with the average value for the 5-minute interval averaged over all the days

```
rowNA<-which(is.na(file[,1]))
calNA<-length(rowNA)
dup_file<-file
dup_file[rowNA,1]<-daily_steps2[match(dup_file[rowNA,3],names(daily_steps2))]
daily_steps3<-sapply(split(dup_file[,1],dup_file[,2]),sum,na.rm=TRUE)
barplot(daily_steps3,xlab="Date", ylab="Number of steps", main="No. of steps taken each day", n
ames.arg=list_dates, cex.lab=1.2)
```




```
m1_3<-mean(daily_steps3)
m2_3<-median(daily_steps3)
```

The mean and meadian of the number of steps per day now is 1.0581×10^4 and 1.0395×10^4, whereas the original mean and median were 9354.2295 and 10395 respectively

The data is segregated according to whether readings correspond to a weekday or a weekend. A panel plot for 5 min interval averaged across all days is plotted separately for each category

```
week_list<-weekdays(as.Date(file[,2],"%Y-%m-%d"))
week_day<-rep("Weekday",length(week_list))
tmp<-which(week_list=="Saturday" | week_list=="Sunday")
week_day[tmp]<-"Weekend"
final_file<-cbind(file,week_day)
wkday<-final_file[final_file[,4]=="Weekday",]
wkend<-final_file[final_file[,4]=="Weekend",]
daily_steps4<-sapply(split(wkday[,1],wkday[,3]),mean,na.rm=TRUE)

daily_steps5<-sapply(split(wkend[,1],wkend[,3]),mean,na.rm=TRUE)

par(mfrow=c(2,1))
plot(x=names(daily_steps4), y=daily_steps4, type="l", main="Average 5-minute interval Activity
Pattern in Weekdays", xlab="5-Minute interval", ylab="Average Steps",ylim=c(0,300), xaxp=c(0,23
55,471),col="red" )
plot(x=names(daily_steps5), y=daily_steps5, type="l", main="Average 5-minute interval Activity
Pattern in Weekends", xlab="5-Minute interval", ylab="Average Steps", xaxp=c(0,2355,471),col="b
lue" )
```