Reproducible Research - Peer Graded Assignment: Course Project 1

Renaud

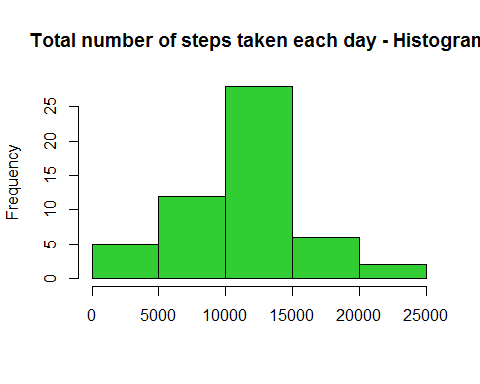
This is the Markdown document addressing the Course Project 1 assignment of the Reproducible Research Course.

Read the activity file with read.csv()

activity<-read.csv("C:/Users/Renaud/Rcourses/reproducibleresearch/week2/project/activity.csv")

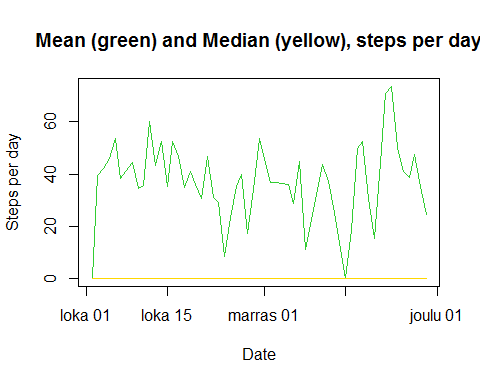
Use aggregate() to aggregate by day and plot the histogram

sum\_steps<-aggregate(activity$steps, list(activity$date), sum)  
hist(sum\_steps$x, main="Total number of steps taken each day - Histogram", col="limegreen", xlab="")



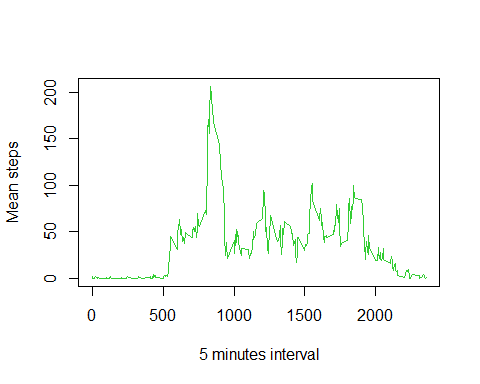
Mean and median per day

activity\_clean<-na.omit(activity)  
mean\_steps<-aggregate(activity\_clean$steps, list(activity\_clean$date), mean)  
mean\_steps$Group.1<-as.Date(mean\_steps$Group.1)  
median\_steps<-aggregate(activity\_clean$steps, list(activity\_clean$date), median)  
median\_steps$Group.1<-as.Date(median\_steps$Group.1)  
  
plot(mean\_steps, type="l", col="limegreen", xlab="Date", ylab="Steps per day", main="Mean (green) and Median (yellow), steps per day")  
lines(median\_steps, type="l", col="gold")



Mean per five minutes interval

mean\_5min<-aggregate(activity\_clean$steps, list(activity\_clean$interval), mean)  
colnames(mean\_5min)<-c("interval","meansteps")  
plot(mean\_5min, col="limegreen", type="l", xlab="5 minutes interval", ylab="Mean steps")



the 5 minutes interval with maximum number of steps is given with code below:

subset(mean\_5min, mean\_5min$meansteps==max(mean\_5min$meansteps))$interval

## [1] 835

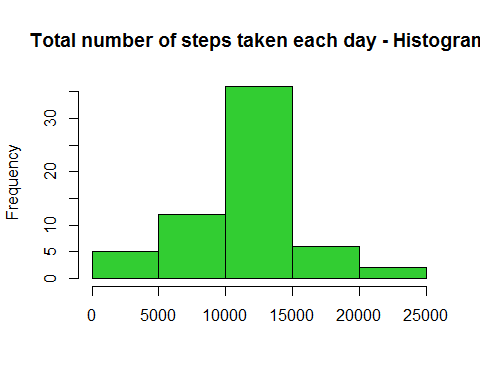
missing values NA calculated with nrow()

nrow(subset(activity, is.na(activity$steps)))

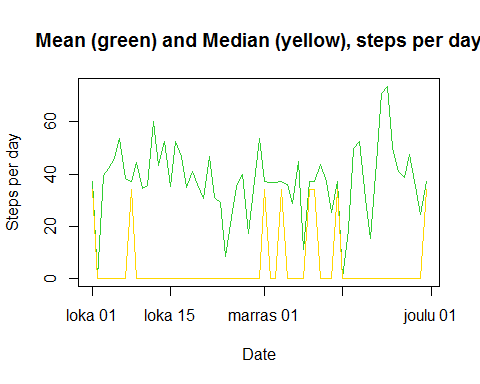
## [1] 2304

We use the mean for that 5minutes interval to fill in the values missing in orginal data set. We plot the new data using same code as above just changing input data set

activity\_mod<-activity  
for (i in 1:nrow(activity\_mod))  
{  
 if (is.na(activity\_mod$steps[i])){  
 step<-subset(mean\_5min, mean\_5min$interval==activity[i,3])$meansteps  
 activity\_mod$steps[i]<-step  
 }  
}  
  
sum\_steps<-aggregate(activity\_mod$steps, list(activity\_mod$date), sum)  
hist(sum\_steps$x, main="Total number of steps taken each day - Histogram", col="limegreen", xlab="")



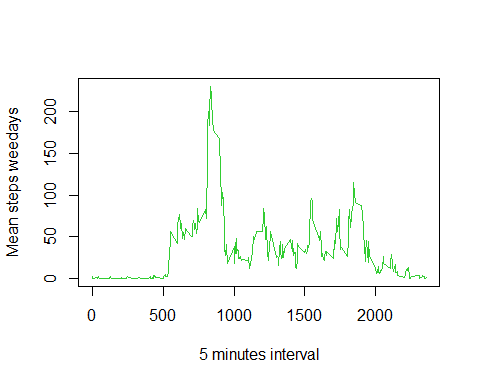
mean\_steps<-aggregate(activity\_mod$steps, list(activity\_mod$date), mean)  
mean\_steps$Group.1<-as.Date(mean\_steps$Group.1)  
median\_steps<-aggregate(activity\_mod$steps, list(activity\_mod$date), median)  
median\_steps$Group.1<-as.Date(median\_steps$Group.1)  
  
plot(mean\_steps, type="l", col="limegreen", xlab="Date", ylab="Steps per day", main="Mean (green) and Median (yellow), steps per day")  
lines(median\_steps, type="l", col="gold")



Comparing those plots with earlier plots, it can be noted that inputing missing data (average of 5minutes interval) can have some impact on the result. Interestingly the average line of steps per day seems quite unchanged.

We now want to compare weekdays and weekends. We first use function weekdays() to provide actual week day. (Note R is providing days in Finnish language... "lauantai" and "sunnuntai" are respectively "saturday" and "sunday")

day<-weekdays(as.Date(activity\_mod$date))  
  
for (i in 1:length(day)){  
 if (day[i]=="lauantai" || day[i]=="sunnuntai")  
 day[i]<-"weekend"  
 else  
 day[i]<-"weekday"  
}  
  
activity\_mod<-cbind(activity\_mod, day)  
weekday<-subset(activity\_mod, activity\_mod$day=="weekday")  
weekend<-subset(activity\_mod, activity\_mod$day=="weekend")  
  
mean\_5min\_wd<-aggregate(weekday$steps, list(weekday$interval), mean)  
colnames(mean\_5min\_wd)<-c("interval","meansteps")  
  
mean\_5min\_we<-aggregate(weekend$steps, list(weekend$interval), mean)  
colnames(mean\_5min\_we)<-c("interval","meansteps")  
  
plot(mean\_5min\_wd, col="limegreen", type="l", xlab="5 minutes interval", ylab="Mean steps weedays")



plot(mean\_5min\_we, col="limegreen", type="l", xlab="5 minutes interval", ylab="Mean steps weekend")

