Peer 1

# Reproducible Research: Peer Assessment 1

1. Load the data (i.e. read.csv()) and process/transform the data into a format suitable for your analysis

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.3.3

library(plyr)

## Warning: package 'plyr' was built under R version 3.3.3

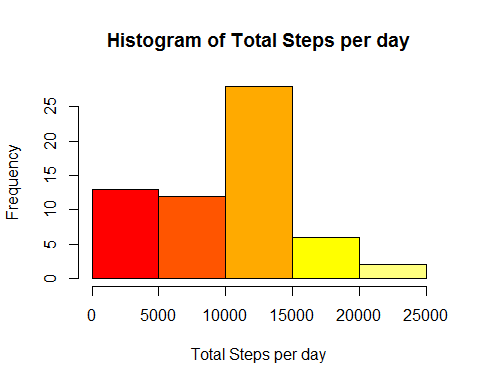
activity <- read.csv("activity.csv",colClasses=c("numeric","Date","numeric"))

# What is mean total number of steps taken per day?

1. Make a histogram of the total number of steps taken each day

Calculate and report the mean and median total number of steps taken per day

sum\_by\_date <- tapply(activity$steps,activity$date,sum,na.rm=TRUE)  
 sum\_steps\_per\_day <- tapply(activity$steps,activity$date,sum,na.rm=TRUE)  
  
  
hist(sum\_steps\_per\_day,col=heat.colors(5),xlab="Total Steps per day",main="Histogram of Total Steps per day")



1. Calculate and report the mean and median total number of steps taken per day

mean(sum\_steps\_per\_day)

## [1] 9354.23

median(sum\_steps\_per\_day)

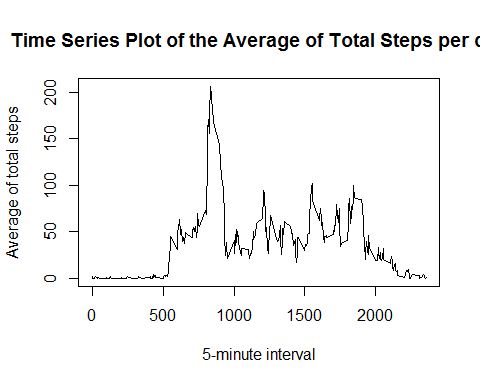
## [1] 10395

# What is the average daily activity pattern?

1. We will make a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

mean\_by\_interval <- tapply(activity$steps,activity$interval,mean,na.rm=TRUE)

plot(row.names(mean\_by\_interval),mean\_by\_interval,type="l",xlab="5-minute interval",ylab="Average of total steps",main="Time Series Plot of the Average of Total Steps per day")



2.Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

x <- max(mean\_by\_interval)  
match(x,mean\_by\_interval)

## [1] 104

mean\_by\_interval[104]

## 835   
## 206.1698

# Imputing missing values

1.Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs) 2.Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc. 3.Create a new dataset that is equal to the original dataset but with the missing data filled in.

sum(is.na(activity))

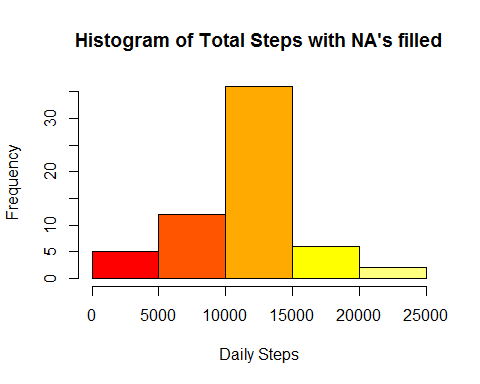
## [1] 2304

activity\_na <- activity[is.na(activity),]  
activity\_no\_na <- activity[complete.cases(activity),]  
activity\_na$steps <- as.numeric(mean\_by\_interval)  
new\_activity <- rbind(activity\_na,activity\_no\_na)  
new\_activity <- new\_activity[order(new\_activity[,2],new\_activity[,3]),]

4.Make a histogram of the total number of steps taken each day and Calculate and report the mean and median total number of steps taken per day.

new\_daily\_sum <- tapply(new\_activity$steps,new\_activity$date,sum)

hist(new\_daily\_sum,col=heat.colors(5),xlab="Daily Steps",main="Histogram of Total Steps with NA's filled")



mean(new\_daily\_sum)

## [1] 10766.19

median(new\_daily\_sum)

## [1] 10766.19

# Are there differences in activity patterns between weekdays and weekends?

1.Create a new factor variable in the dataset with two levels -- "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.

days <- weekdays(new\_activity[,2])  
new\_activity <- cbind(new\_activity,days)  
  
library(plyr)  
  
new\_activity$days <- revalue(new\_activity$days,c("Δευτέρα"="weekday","Τρίτη"="weekday","Τετάρτη"="weekday","Πέμπτη"="weekday","Παρασκευή"="weekday"))  
  
new\_activity$days <- revalue(new\_activity$days,c("Σάββατο"="weekend","Κυριακή"="weekend"))  
  
  
new\_mean\_interval <- tapply(new\_activity$steps,list(new\_activity$interval,new\_activity$days),mean)  
  
library(reshape2)

## Warning: package 'reshape2' was built under R version 3.3.3

new\_mean\_interval <- melt(new\_mean\_interval)  
colnames(new\_mean\_interval) <- c("interval","day","steps")  
library(lattice)

2.Make a panel plot containing a time series plot (i.e. type = "l") of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all weekday days or weekend days (y-axis)

xyplot(new\_mean\_interval$steps ~ new\_mean\_interval$interval | new\_mean\_interval$day, layout=c(1,2),type="l",main="Time Series Plot of the Average of Total Steps (weekday/weekend)",xlab="Time intervals",ylab="Average Total Steps")

