# Course Project 1 Reproducible Research Tony Schollum

## Step 1 Read in the dataset

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

set the environment

Explore the data

head(activity)

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

summary(activity)

## steps date interval   
## Min. : 0.00 2012-10-01: 288 Min. : 0.0   
## 1st Qu.: 0.00 2012-10-02: 288 1st Qu.: 588.8   
## Median : 0.00 2012-10-03: 288 Median :1177.5   
## Mean : 37.38 2012-10-04: 288 Mean :1177.5   
## 3rd Qu.: 12.00 2012-10-05: 288 3rd Qu.:1766.2   
## Max. :806.00 2012-10-06: 288 Max. :2355.0   
## NA's :2304 (Other) :15840

It appears that we have to change the date column into a date format, the best option here is lubridate

activity$date<-ymd(activity$date)  
length(unique(activity$date))

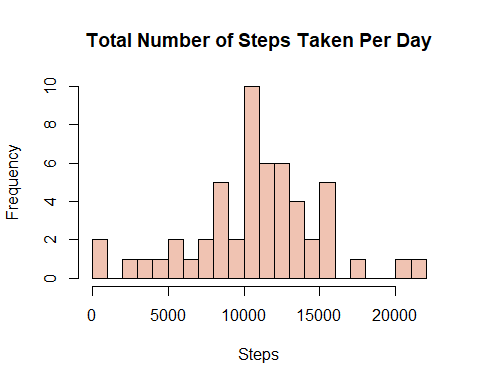
## [1] 61

and there are a number of missing steps counts to fix

activity\_na<-subset(activity, !is.na(activity$steps))

## Step 2 Histogram of Total Number of Steps taken per Day

SPD<-tapply(activity\_na$steps, activity\_na$date, sum, na.rm=TRUE, simplify = T)  
  
hist(x = SPD, col = rgb(0.8, 0.22,0,0.3), breaks = 30, xlab = "Steps", ylab = "Frequency", main = "Total Number of Steps Taken Per Day")



## Step 3 Mean and Median Steps per Day

Use of simple functions

mean(SPD)

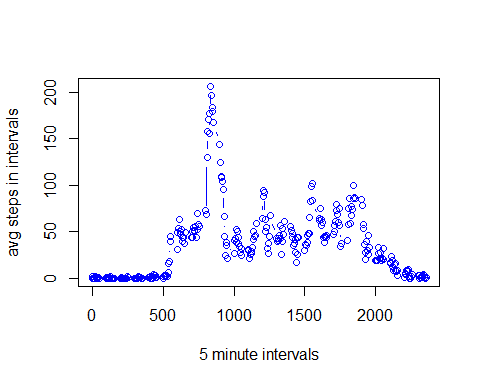
## [1] 10766.19

median(SPD)

## [1] 10765

## Step 4 Average Daily Activity Pattern

Interval<- tapply(activity\_na$steps, activity\_na$interval, mean, na.rm = TRUE, simplify = T)  
Int <- data.frame(interval = as.integer(names(Interval)), avg = Interval)  
with(Int,  
 plot(interval,  
 avg,  
 type = "b",  
 col = "blue",  
 xlab = "5 minute intervals",  
 ylab = "avg steps in intervals"))



## Step 5, Interval showing the maximum number of steps```{r}

From the plot it appears to be around 800

max\_steps <- max(Int$avg)  
Int[Int$avg ==max\_steps,]

## interval avg  
## 835 835 206.1698

## Step 6, Code to impute missing data

sum(is.na(activity$steps))

## [1] 2304

To calculate the missing data we can use the mean for the interval that is missing

activity$CompleteSteps <- ifelse(is.na(activity$steps), round(activity$steps[match(activity$interval, activity$interval)],0), activity$steps)

activityImp <- data.frame(steps=activity$CompleteSteps, interval=activity$interval, date=activity$date)  
head(activityImp)

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

Step 7 use Weekday functionto see if there are differences in weekends to weekdays

WW <- mutate(activity, weektype = ifelse(weekdays(activity$date) == "Saturday" | weekdays(activity$date) == "Sunday", "weekend", "weekday"))

## Warning: package 'bindrcpp' was built under R version 3.5.1

WW$weektype <- as.factor(WW$weektype)

Calculate 5 min intervals and use ggplot

interval\_full <- WW %>% group\_by(interval, weektype)   
FM <- ggplot(interval\_full, aes(x=interval, y=steps, color = weektype)) + geom\_line() + facet\_wrap(~weektype, ncol = 1, nrow=2)  
 print(FM)

## Warning: Removed 2 rows containing missing values (geom\_path).

