Reproducible Research: Peer Assessment 1

Loading and preprocessing the data

Below, the data are read using read.csv, then incomplete cases are eliminated and I split the data frame using split by $date.

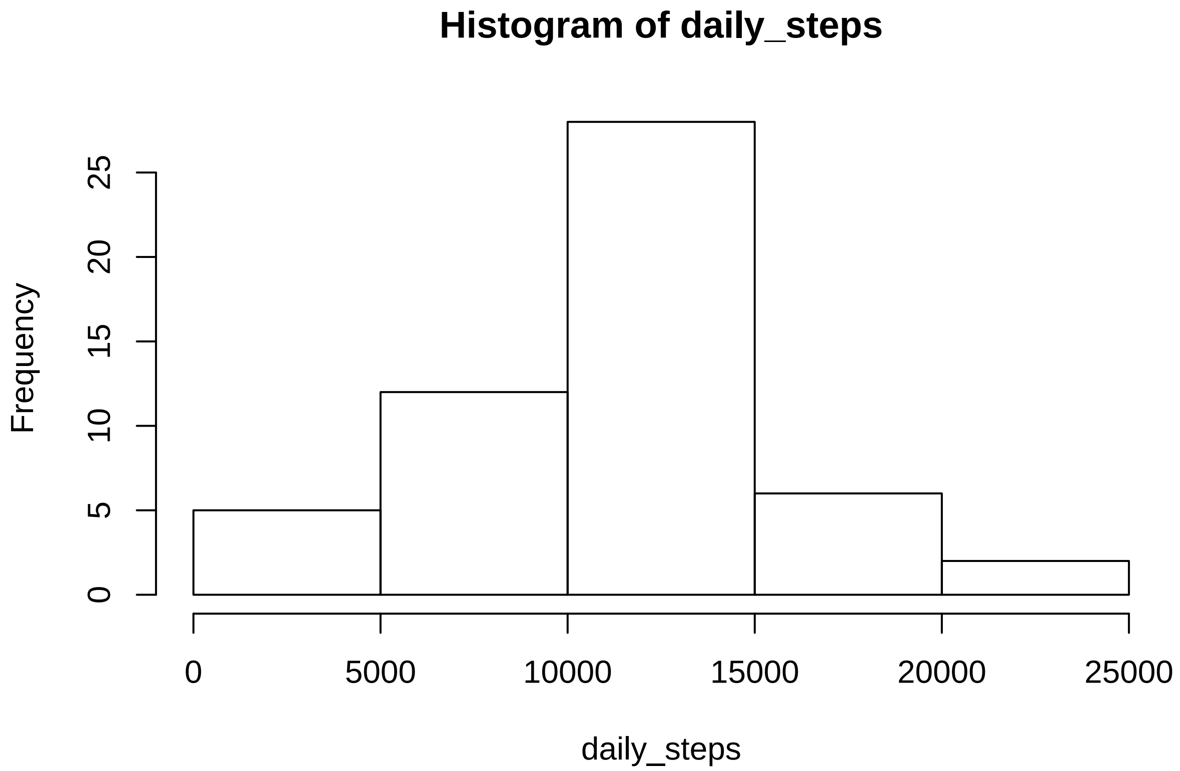
```
par(mfrow=c(1,1))
options(scipen=4)
activity_filename <- "activity.csv"
act <- read.csv(activity_filename, stringsAsFactors=FALSE)
complete_act <- act[complete.cases(act),]
bydate_act <- split(complete_act, complete_act$date, drop=TRUE)
coltest <- bydate_act[[1]][1:3,1:3]
print(coltest)
```

```
##     steps       date interval
## 289     0 2012-10-02        0
## 290     0 2012-10-02        5
## 291     0 2012-10-02       10
```

What is mean total number of steps taken per day?

This code calculates the mean # of steps taken per day by taking the split data sets and looping over them to sum the steps per day and then append to a list, then takes the mean & median of the list.

```
c <- complete_act #lazeh
a <- bydate_act  #lazy
daily_steps <- vector()
for(date in bydate_act){
 daily_steps <- append(daily_steps, as.numeric(sum(date$steps)))
#  mean(bydate_act[["2012-11-29"]]$"steps")      # just for future use, if necessary
}
myhist <- hist(daily_steps)
```

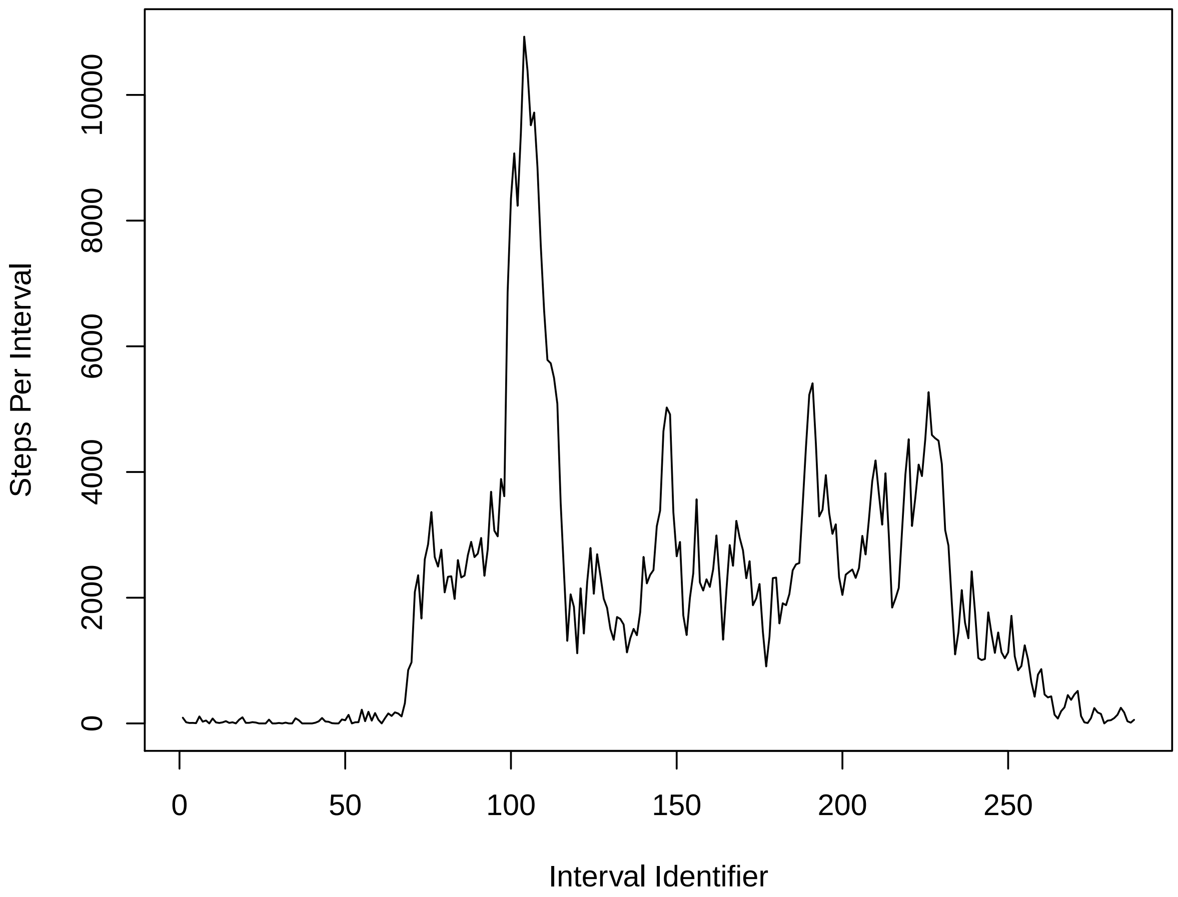


```
daily_mean <- mean(daily_steps)
daily_median <- median(daily_steps)
```

The mean of the per-day steps is 10766.1887
The median of the per-day steps is 10765

What is the average daily activity pattern?

```
by_interval_act <- split(complete_act, complete_act$interval, drop=TRUE)

interval_steps <- vector()
for(interval in by_interval_act){
  interval_steps <- append(interval_steps, sum(interval$steps))
}

plot(interval_steps, type="l", ylab = "Steps Per Interval", xlab = "Interval Identifier")
```



```
max_interval <- which.max(interval_steps)
```

The interval with the most steps is interval 104

Imputing missing values

Calculating the # of cases containing NA values. I

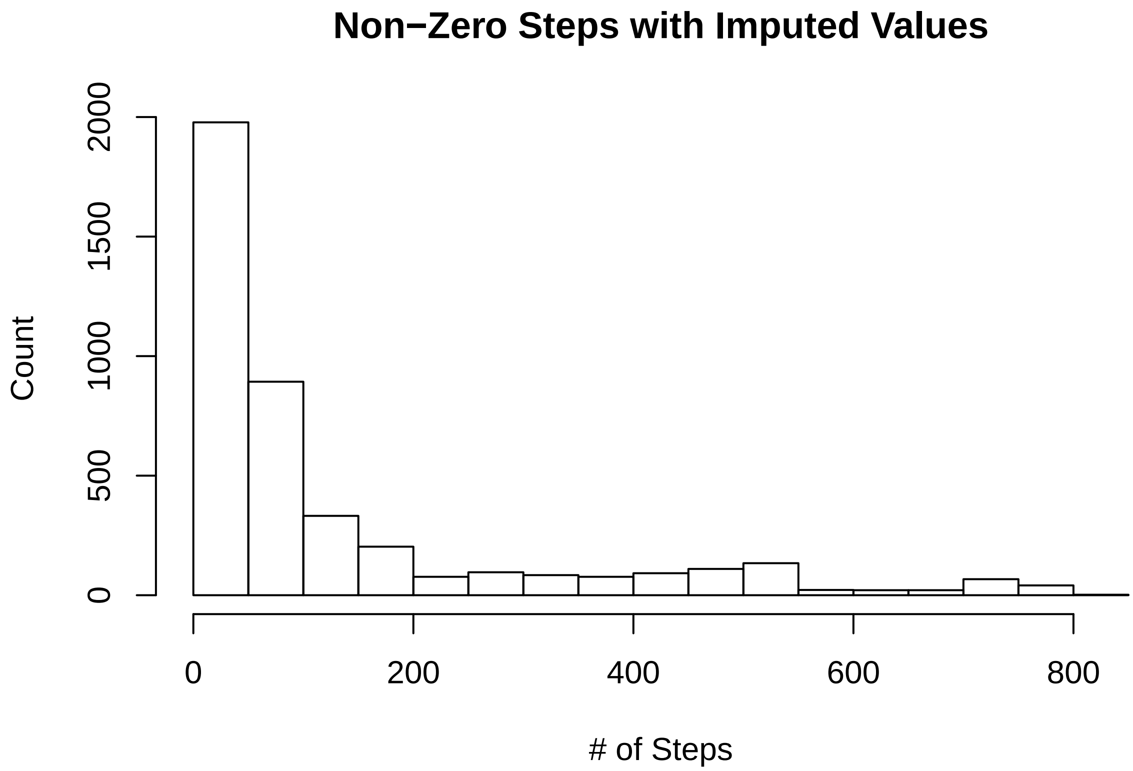
```
total_count <- length(act$date)
temp <- act[complete.cases(act),]
complete_count <- length(temp$date)
no_of_na <- total_count - complete_count
```

I just took the total # of cases and subtracted the # of complete cases to get the total # of cases containing NA's.

The number of cases containing an NA is 2304

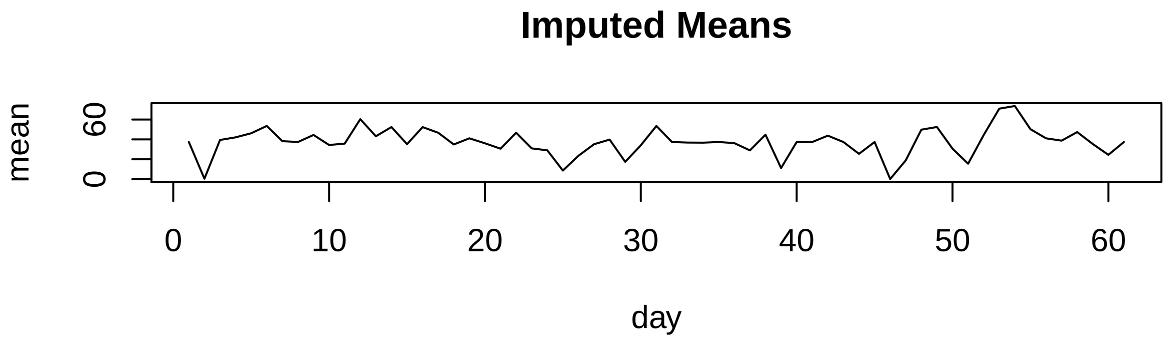
The strategy for filling in the missing blanks is to take the mean of each interval and then match up the interval that's NA and replace with the interval mean for that interval. There are still a lot of 0's, even after replacing the NA values, so in order show the difference,

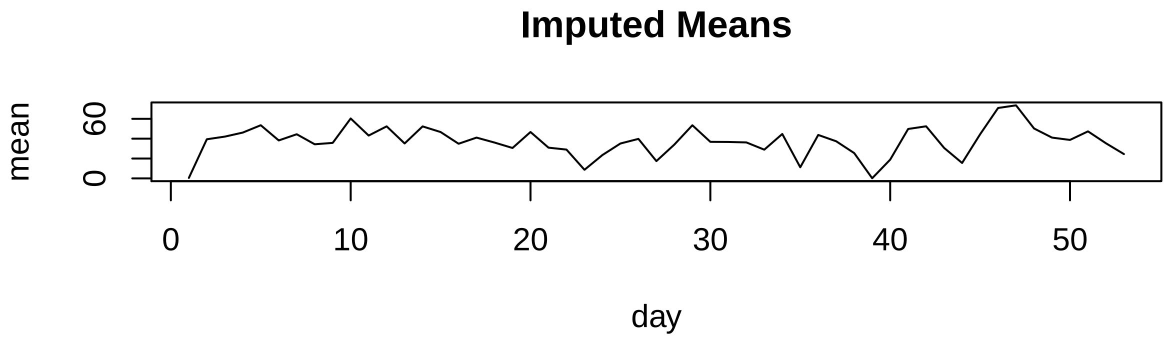
```
na_act <- act
interval_avg <- aggregate(act$steps ~ act$interval, FUN=mean)
names(interval_avg) <- c("interval", "avsteps")
ia <- interval_avg
for(i in 1:nrow(na_act)){
  if(is.na(na_act[i,1]) == TRUE){
    na_act[i,1] <- ia[ia$interval == na_act[i,3],]$avsteps

    }
}
hist(na_act$steps[act$steps!=0], main="Non-Zero Steps with Imputed Values", xlab="# of Steps", ylab="Co
```



Now, the assignment says to do this: Calculate and report the mean and median total number of steps taken per day. Do these values differ from the estimates from the first part of the assignment? What is the impact of imputing missing data on the estimates of the total daily number of steps?

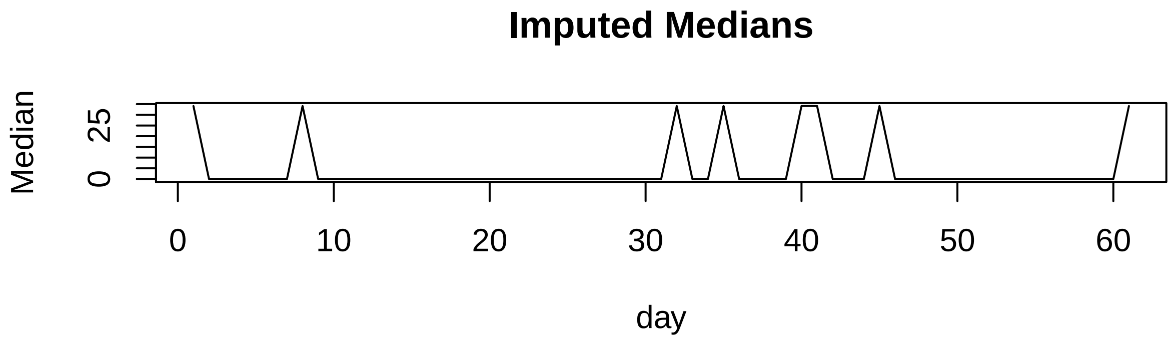
```
na_means <- aggregate(na_act$steps ~ na_act$date, FUN=mean)
orig_means <- aggregate(act$steps ~ act$date, FUN=mean)
par(mfrow=c(2,1))
plot(na_means[,2], type="l",main="Imputed Means", xlab="day", ylab="mean")
plot(orig_means[,2], type="l", main="Imputed Means", xlab="day", ylab="mean")
```

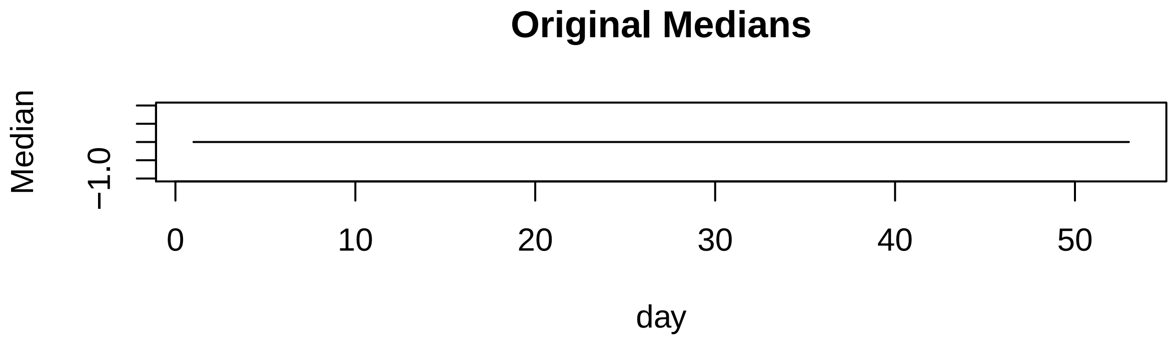




Now the medians:

```
na_medians <- aggregate(na_act$steps ~ na_act$date, FUN=median)
orig_medians <- aggregate(act$steps ~ act$date, FUN=median, na.rm=TRUE)
par(mfrow=c(2,1))
plot(na_medians[,2], type="l", main="Imputed Medians", xlab="day", ylab="Median")
plot(orig_medians[,2], type="l", main="Original Medians", xlab="day", ylab="Median")
```

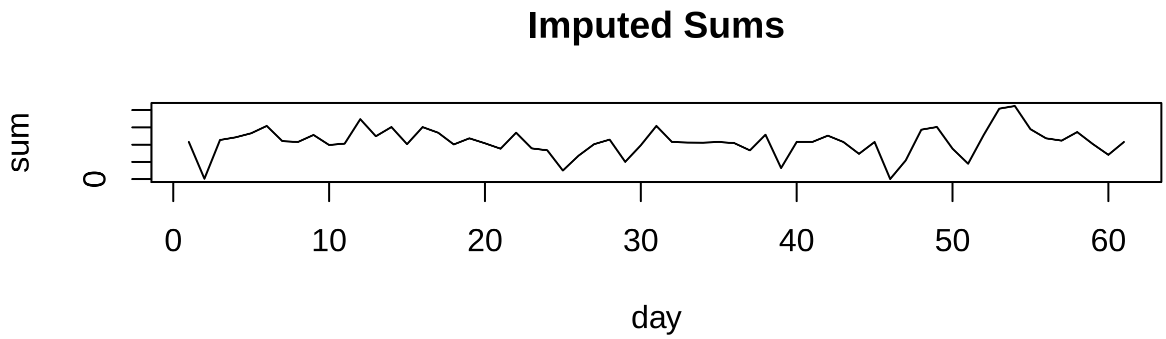


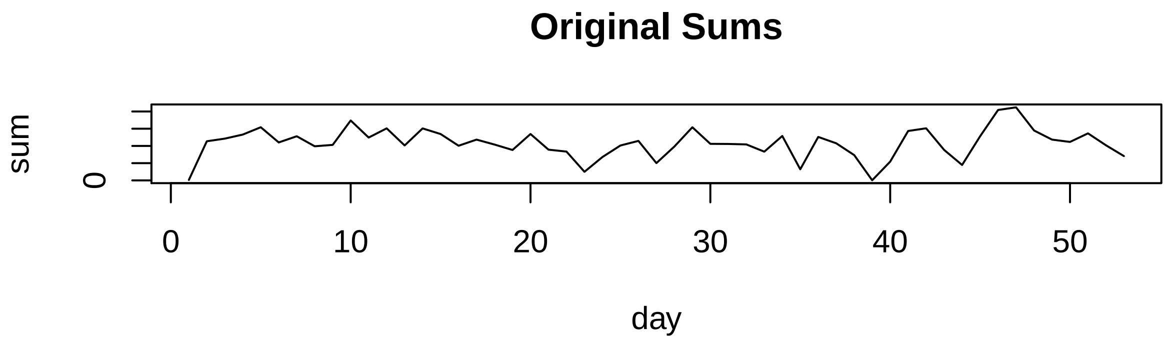


So the means barely changed, but the medians look significantly different. This doesn't seem right, since the median is just the centric values of the distribution, but I've been over it a few times and can't seem to get a resolution, so I'll stick with these results for now – perhaps I'll take another look after the rest is complete.

Regarding the impact on the total # of steps per day, these two graphs illustrate the differences:

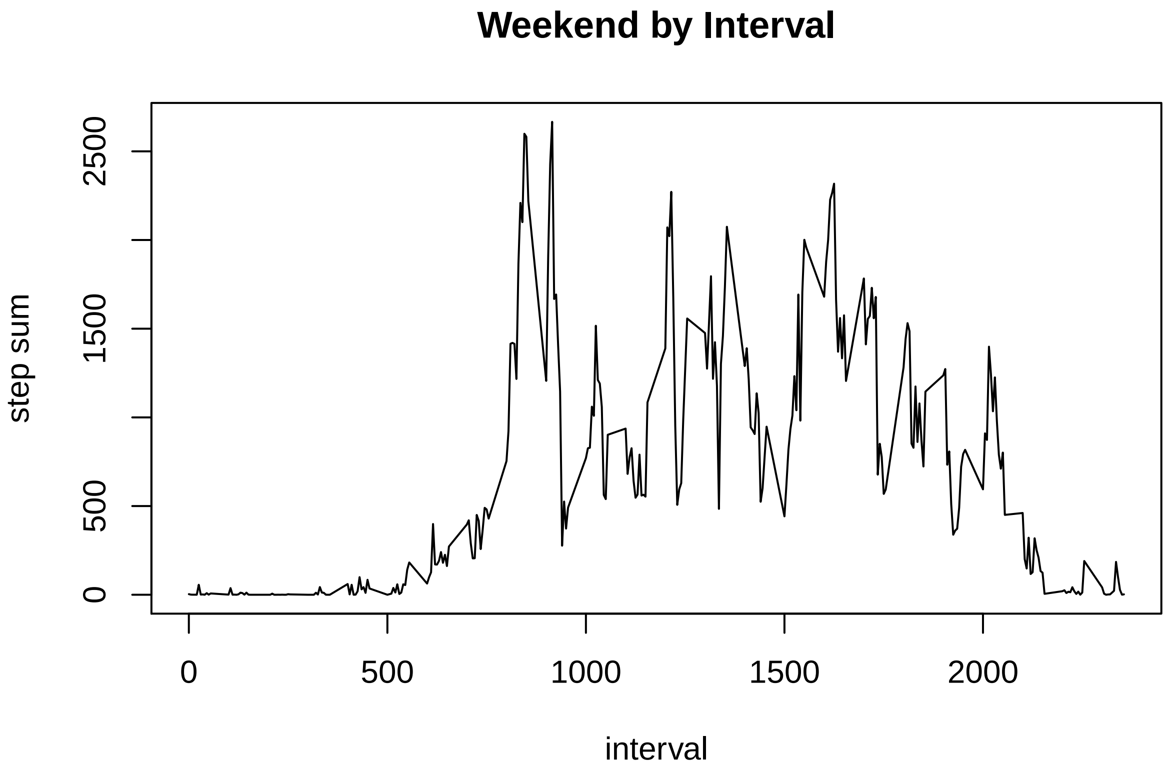
```r
na_sums <- aggregate(na_act$steps ~ na_act$date, FUN=sum)
orig_sums <- aggregate(act$steps ~ act$date, FUN=sum)
par(mfrow=c(2,1))
plot(na_sums[,2], type="l", main="Imputed Sums", xlab="day", ylab="sum")
plot(orig_sums[,2], type="l", main="Original Sums", xlab="day", ylab="sum")
```



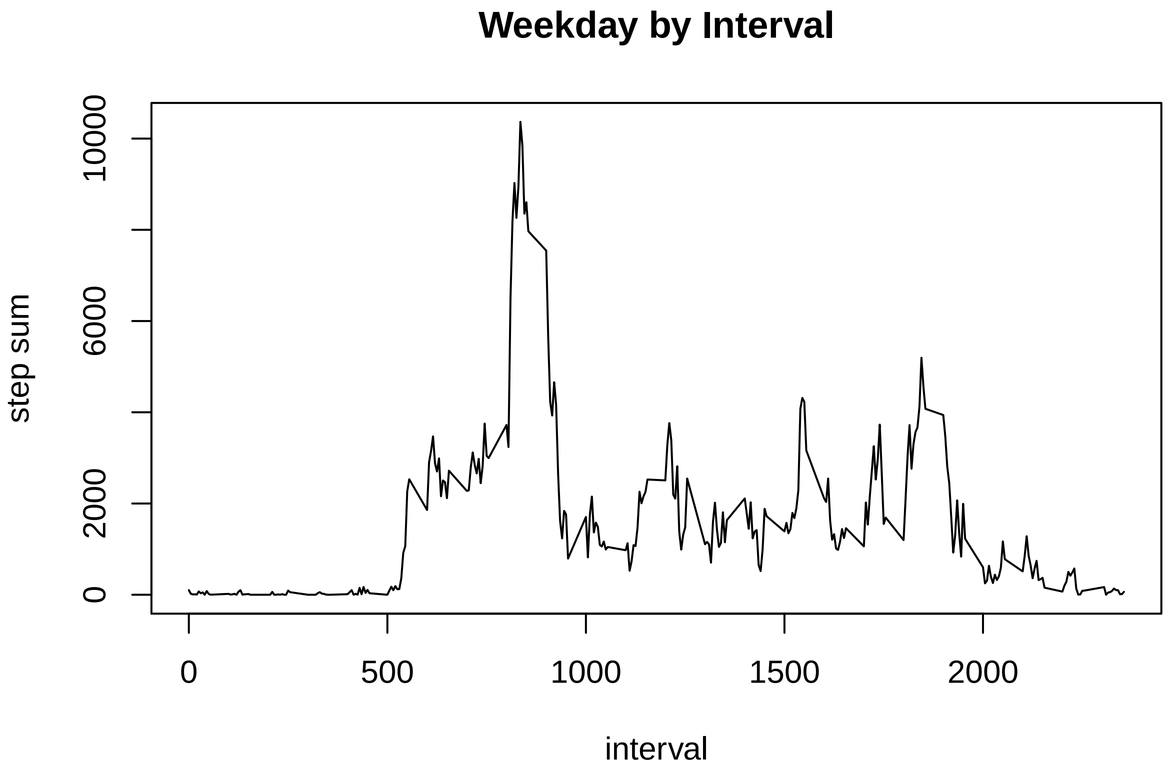


Are there differences in activity patterns between weekdays and weekends?

```
# wknd <- act
# wkdy <- act
# wknd$date <- weekdays(as.POSIXct(act$date))
# wkdy$date <- weekdays(as.POSIXct(act$date))
wkact <- na_act
wkact$day <- weekdays(as.POSIXct(act$date))
wkact$daytype[wkact$day %in% c("Saturday", "Sunday")] <- "weekend"
wkact$daytype[wkact$day %in% c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")] <- "weekday"
# cbind(wknd, wknd[wknd$date %in% c("Saturday", "Sunday"),]])
# wkdy <- wkdy[wkdy$date %in% c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"),]
wknd_by_interval <- aggregate(wkact[wkact$daytype=="weekend",1] ~ wkact[wkact$daytype=="weekend", 3], FU
plot(wknd_by_interval, type="l", main = "Weekend by Interval", xlab="interval", ylab = "step sum")
```



```
wkdy_by_interval <- aggregate(wkact[wkact$daytype=="weekday",1] ~ wkact[wkact$daytype=="weekday", 3], F
plot(wkdy_by_interval, type="l", main = "Weekday by Interval", xlab="interval", ylab = "step sum")
```



Yes, there is more activity in the later intervals during the weekend (probably during work hours) and less activity on the weekdays.