Assessment 1

Ruben Adad

November 15, 2014

Loading and preprocessing the data

```
setwd("~/Documents/CURSOS/Reproducible Research/RepData_PeerAssessment1")
activity <- read.csv("activity.csv", stringsAsFactors=F)
summary(activity)
```

```
##      steps            date              interval     
##  Min.   :  0.00   Length:17568       Min.   :   0.0  
##  1st Qu.:  0.00   Class :character   1st Qu.: 588.8  
##  Median :  0.00   Mode  :character   Median :1177.5  
##  Mean   : 37.38                      Mean   :1177.5  
##  3rd Qu.: 12.00                      3rd Qu.:1766.2  
##  Max.   :806.00                      Max.   :2355.0  
##  NA's   :2304
```

```
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
```

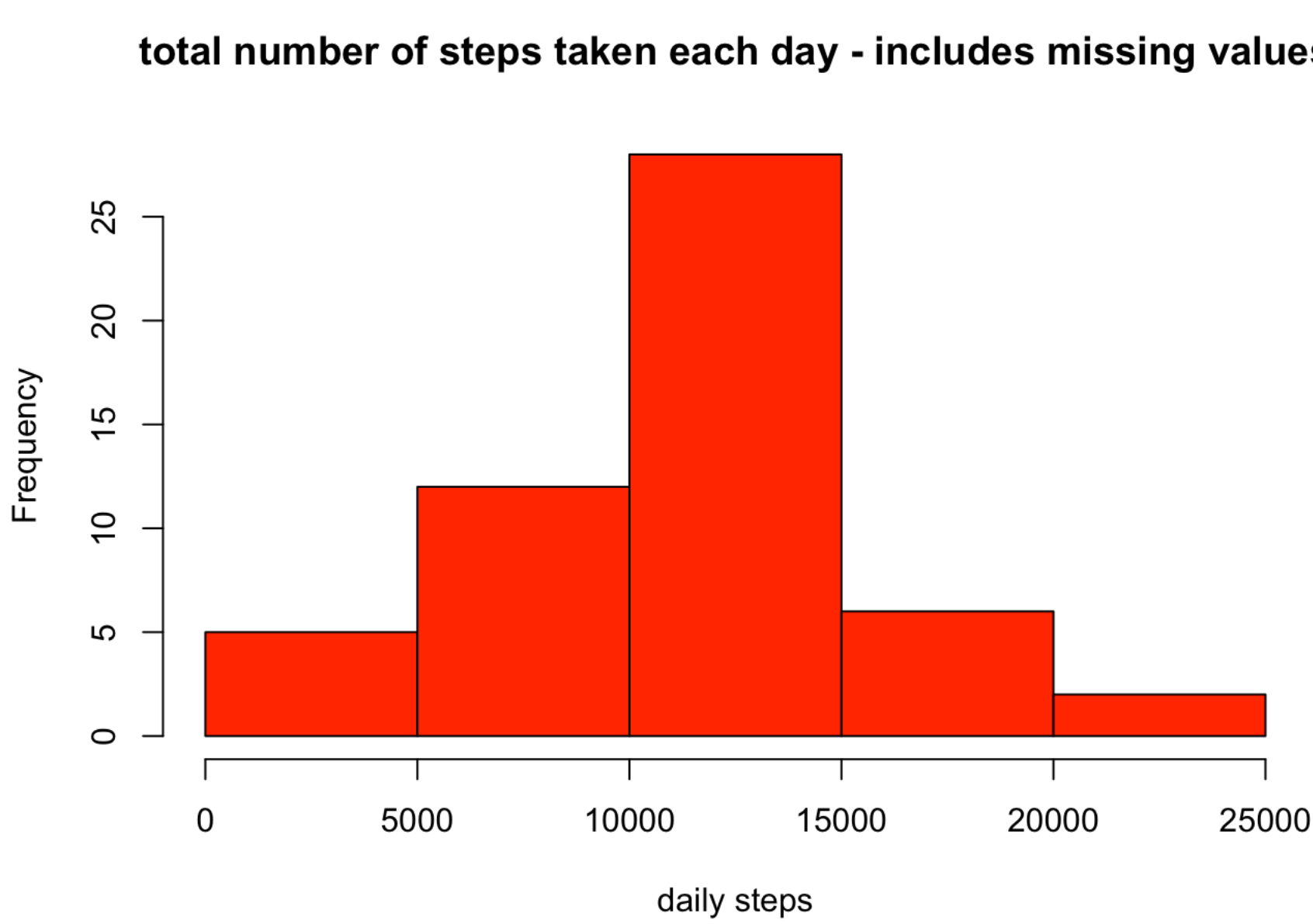
What is mean total number of steps taken per day?

The following histogram shows the distribution of the total number of steps taken each day. The distribution **includes missing values**.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
grp <- group_by(activity, date)
tot_steps <- summarise(grp, Total_steps = sum(steps))
tot_steps$date <- as.Date(tot_steps$date, "%Y-%m-%d")
hist(tot_steps$Total_steps, main="total number of steps taken each day - includes missing v
alues", xlab="daily steps", col="red")
```




```
avg_steps <- summarise(grp, Mean = mean(steps), Median = median(steps))
```

The mean and median total number of steps taken per day including missing values

```
library(xtable)
print(xtable(as.data.frame(avg_steps), caption="mean and median steps taken per day"), html
.table.attributes = list('border="2" bordercolor="blue" style=width:50%'), type="html")
```

	date	Mean	Median
1	2012-10-01		
2	2012-10-02	0.44	0
3	2012-10-03	39.42	0
4	2012-10-04	42.07	0
5	2012-10-05	46.16	0
6	2012-10-06	53.54	0
7	2012-10-07	38.25	0
8	2012-10-08		
9	2012-10-09	44.48	0
10	2012-10-10	34.38	0
11	2012-10-11	35.78	0
12	2012-10-12	60.35	0
13	2012-10-13	43.15	0
14	2012-10-14	52.42	0
15	2012-10-15	35.20	0
16	2012-10-16	52.38	0
17	2012-10-17	46.71	0
18	2012-10-18	34.92	0
19	2012-10-19	41.07	0
20	2012-10-20	36.09	0
21	2012-10-21	30.63	0
22	2012-10-22	46.74	0
23	2012-10-23	30.97	0
24	2012-10-24	29.01	0
25	2012-10-25	8.65	0
26	2012-10-26	23.53	0
27	2012-10-27	35.14	0
28	2012-10-28	39.78	0
29	2012-10-29	17.42	0
30	2012-10-30	34.09	0
31	2012-10-31	53.52	0

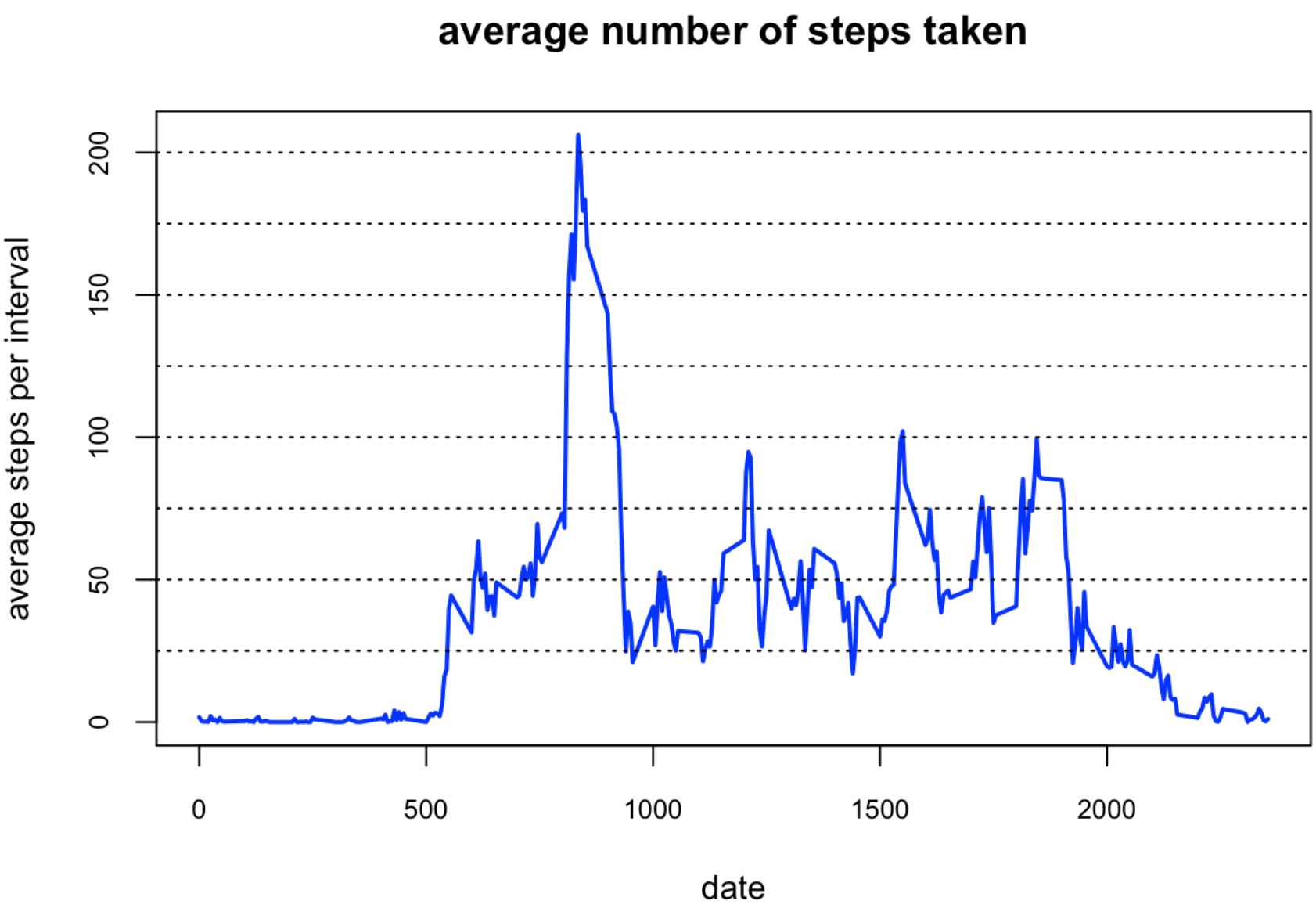
32	2012-11-01		
33	2012-11-02	36.81	0
34	2012-11-03	36.70	0
35	2012-11-04		
36	2012-11-05	36.25	0
37	2012-11-06	28.94	0
38	2012-11-07	44.73	0
39	2012-11-08	11.18	0
40	2012-11-09		
41	2012-11-10		
42	2012-11-11	43.78	0
43	2012-11-12	37.38	0
44	2012-11-13	25.47	0
45	2012-11-14		
46	2012-11-15	0.14	0
47	2012-11-16	18.89	0
48	2012-11-17	49.79	0
49	2012-11-18	52.47	0
50	2012-11-19	30.70	0
51	2012-11-20	15.53	0
52	2012-11-21	44.40	0
53	2012-11-22	70.93	0
54	2012-11-23	73.59	0
55	2012-11-24	50.27	0
56	2012-11-25	41.09	0
57	2012-11-26	38.76	0
58	2012-11-27	47.38	0
59	2012-11-28	35.36	0
60	2012-11-29	24.47	0
61	2012-11-30		

mean and median steps taken per day

What is the average daily activity pattern?

The following line chart shows the average number of steps taken per 5 minute interval. It also **includes missing values**.

```
grp <- group_by(activity, interval)
avg_interval <- summarise(grp, Mean = mean(steps, na.rm=T), Median = median(steps, na.rm=T)
)
plot(avg_interval$interval, avg_interval$Mean, type="l", lwd=2, col="blue", main="average n
umber of steps taken", xlab="date", ylab="average steps per interval", cex.axis=0.8)
abline(h=c(25,50,75,100,125,150,175,200), lty=3)
```




```
max_interval <- subset(avg_interval, avg_interval$Mean == max(na.omit(avg_interval$Mean)),
select=c(interval, Mean))
```

The 5-minute interval with the maximum number of steps is: **835** with mean **206.1698113**.

Imputing missing values

First I check for missing values in all the rows of the *activity* dataset.

```
cc <- table(complete.cases(activity))
missing_values <- cc["FALSE"]
```

The total number of rows with missing values is: 2304
Then I calculate the mean of all the non-missing values of the *steps* variable. And assign it to all the missing values of de *Mean* attribute in the *avg_steps* data frame.
Next I paste the *avg_steps* data frame to the *activity* data frame using a left join by *date*.
Finally, I assign the *Mean* variable from *avg_steps* to the missing values of the *steps* variable in the *activity* data frame and drop the *Mean* and *Median* columns.

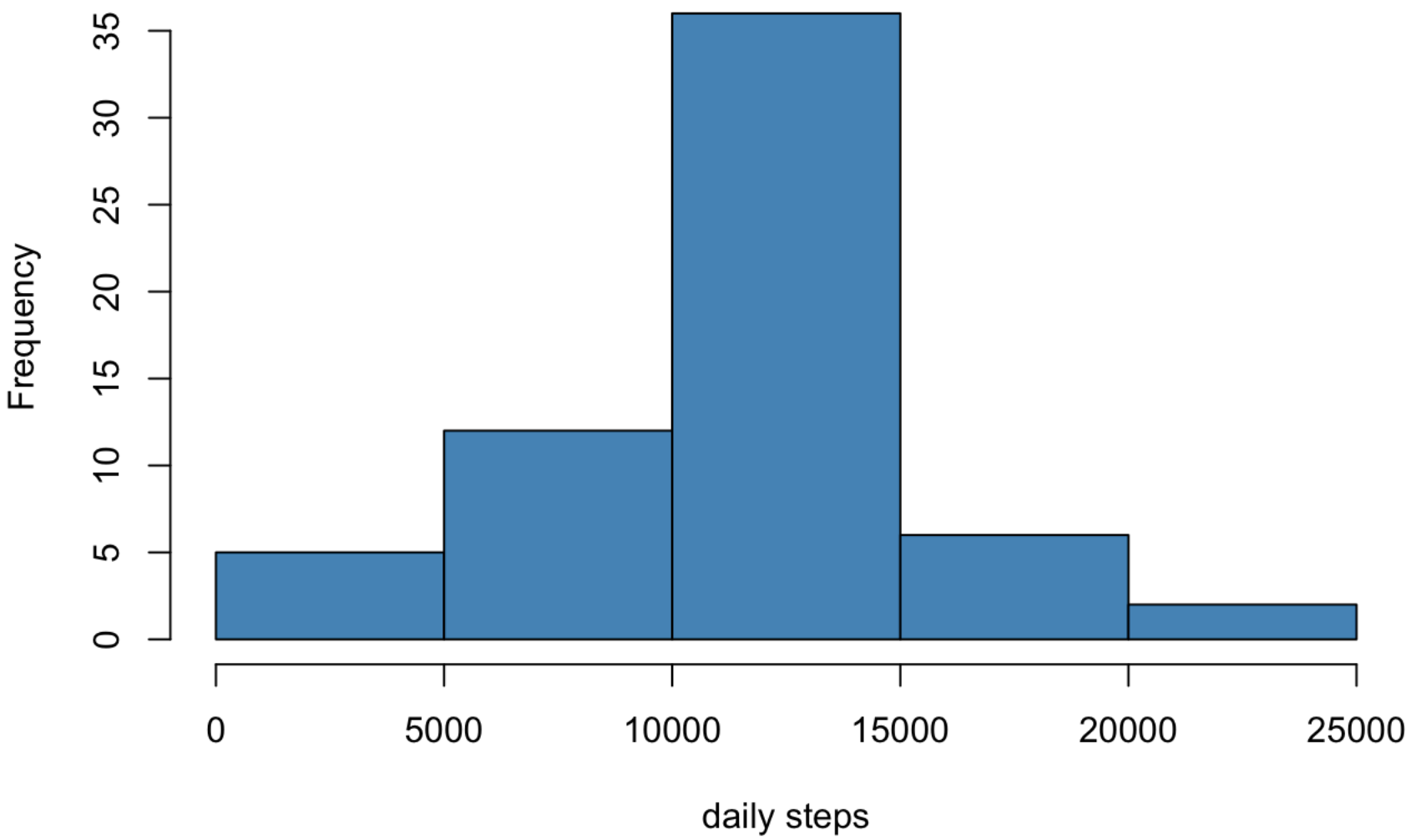
```
avg_steps$Mean[which(complete.cases(avg_steps) %in% "FALSE")] <- mean(na.omit(activity$step
s))
new_activity <- left_join(activity, avg_steps)
```

```
## Joining by: "date"
```

```
new_activity$steps[which(complete.cases(new_activity$steps) %in% "FALSE")] <- new_activity$
Mean[which(complete.cases(new_activity$steps) %in% "FALSE")]
new_activity <- subset(new_activity, select=c("date", "steps", "interval"))
```

Now I repeat the sum of *steps* group by date to create the histogram **without missing values**. And then to print the mean and median of the number of steps per day.
As can be seen this result is different from the previous one which includes missing values.

```
grp <- group_by(new_activity, date)
tot_steps <- summarise(grp, Total_steps = sum(steps))
tot_steps$date <- as.Date(tot_steps$date, "%Y-%m-%d")
hist(tot_steps$Total_steps, main="total number of steps taken each day - excluding missing
values", xlab="daily steps", col="steelblue")
```




```
avg_steps <- summarise(grp, Mean = mean(steps), Median = median(steps))
```

The mean and median total number of steps taken per day excluding missing values

```
print(xtable(as.data.frame(avg_steps), caption="mean and median steps taken per day"), html
.table.attributes = list('border="2" bordercolor="blue" style=width:50%'), type="html")
```

	date	Mean	Median
1	2012-10-01	37.38	37.38
2	2012-10-02	0.44	0.00
3	2012-10-03	39.42	0.00
4	2012-10-04	42.07	0.00
5	2012-10-05	46.16	0.00
6	2012-10-06	53.54	0.00
7	2012-10-07	38.25	0.00

8	2012-10-08	37.38	37.38
9	2012-10-09	44.48	0.00
10	2012-10-10	34.38	0.00
11	2012-10-11	35.78	0.00
12	2012-10-12	60.35	0.00
13	2012-10-13	43.15	0.00
14	2012-10-14	52.42	0.00
15	2012-10-15	35.20	0.00
16	2012-10-16	52.38	0.00
17	2012-10-17	46.71	0.00
18	2012-10-18	34.92	0.00
19	2012-10-19	41.07	0.00
20	2012-10-20	36.09	0.00
21	2012-10-21	30.63	0.00
22	2012-10-22	46.74	0.00
23	2012-10-23	30.97	0.00
24	2012-10-24	29.01	0.00
25	2012-10-25	8.65	0.00
26	2012-10-26	23.53	0.00
27	2012-10-27	35.14	0.00
28	2012-10-28	39.78	0.00
29	2012-10-29	17.42	0.00
30	2012-10-30	34.09	0.00
31	2012-10-31	53.52	0.00
32	2012-11-01	37.38	37.38
33	2012-11-02	36.81	0.00
34	2012-11-03	36.70	0.00
35	2012-11-04	37.38	37.38
36	2012-11-05	36.25	0.00
37	2012-11-06	28.94	0.00
38	2012-11-07	44.73	0.00
39	2012-11-08	11.18	0.00
40	2012-11-09	37.38	37.38
41	2012-11-10	37.38	37.38
42	2012-11-11	43.78	0.00
43	2012-11-12	37.38	0.00
44	2012-11-13	25.47	0.00
45	2012-11-14	37.38	37.38
46	2012-11-15	0.14	0.00
47	2012-11-16	18.89	0.00

48	2012-11-17	49.79	0.00
49	2012-11-18	52.47	0.00
50	2012-11-19	30.70	0.00
51	2012-11-20	15.53	0.00
52	2012-11-21	44.40	0.00
53	2012-11-22	70.93	0.00
54	2012-11-23	73.59	0.00
55	2012-11-24	50.27	0.00
56	2012-11-25	41.09	0.00
57	2012-11-26	38.76	0.00
58	2012-11-27	47.38	0.00
59	2012-11-28	35.36	0.00
60	2012-11-29	24.47	0.00
61	2012-11-30	37.38	37.38

mean and median steps taken per day

Are there differences in activity patterns between weekdays and weekends?

I use the function “isWeekend” from the “timeDate” package to determine if a date is weekend or weekday.
Then I replace the logical values TRUE/FALSE with weekend/weekday respectively.
I generate the comparative graph using ggplot.
As can be seen from the red trendline, weekend and weekdays have different patterns.

```
library(timeDate)
```

```
## Warning: package 'timeDate' was built under R version 3.1.2
```

```
##
## Attaching package: 'timeDate'
##
## The following object is masked from 'package:xtable':
##
##     align
```

```
library(ggplot2)
new_activity$weekend <- factor(isWeekend(as.Date(new_activity$date)))
new_activity$weekend <- gsub("TRUE", "weekend", new_activity$weekend)
new_activity$weekend <- gsub("FALSE", "weekday", new_activity$weekend)
grp <- group_by(new_activity, weekend, interval)
avg_interval <- summarise(grp, Mean = mean(steps), Median = median(steps))
g <- ggplot(avg_interval, aes(interval, Mean))
g + geom_line(color="steelblue") + facet_grid(weekend ~ .) + labs(y = "average steps per in
terval") + geom_smooth(method="lm", color="red")
```