Assignment2-markdown

Chengwen Luo

17/03/2018

Loading and preprocessing the data

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
dat<-read.csv("activity.csv",sep=",",header=T)
dat_org<-dat
head(dat)
```

```
##   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
```

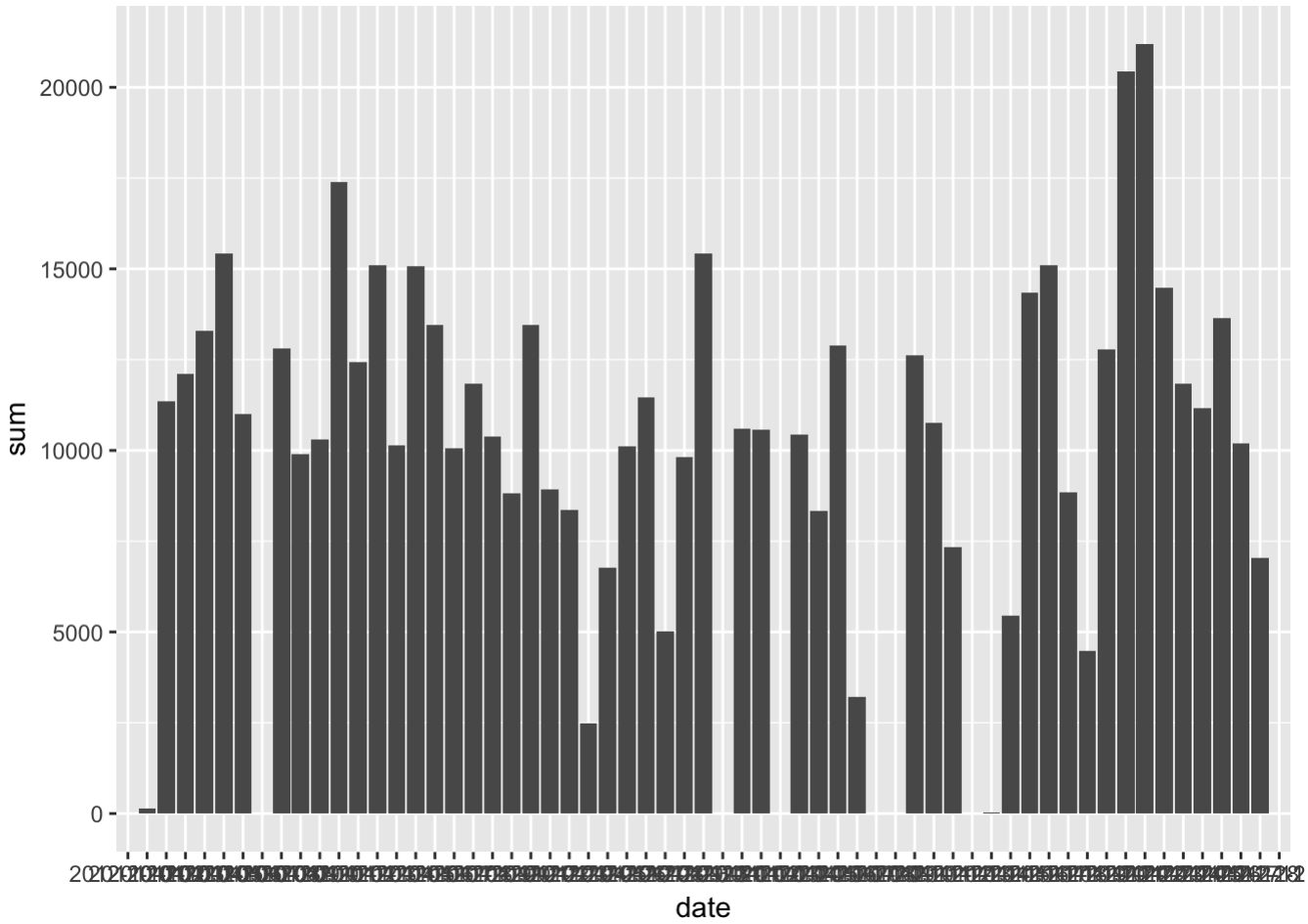
2. Histogram of the total number of steps taken each day

```
tot_step<-as.data.frame(tapply(dat$steps,dat$date,sum))
date<-rownames(tot_step)
rownames(tot_step)=1:dim(tot_step)[1]
tot_step[,2]<-date
names(tot_step)<-c("sum","date")
```

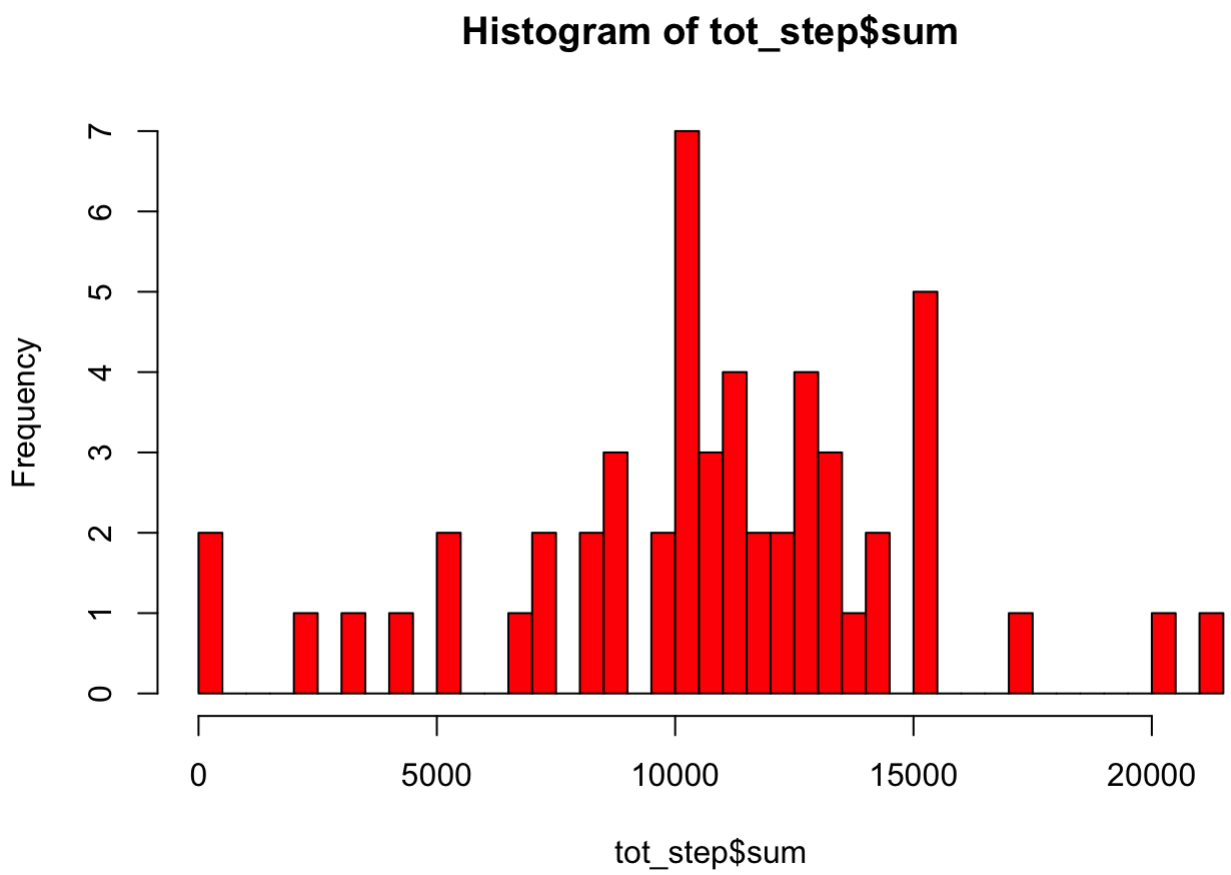
First is a time series bar chat, second one is a histgram

```
g<-ggplot(tot_step,aes(y=sum,x=date))
g+geom_bar(stat="identity")
```

```
## Warning: Removed 8 rows containing missing values (position_stack).
```




```
hist(tot_step$sum,breaks=61,col="red")
```



Mean and median number of steps taken each day

```
dat$date<-ymd(as.vector(dat$date))
median_step<-tapply(dat$steps,dat$date,median,na.rm=T)
mean_step<-tapply(dat$steps,dat$date,mean,na.rm=T)
head(as.data.frame(median_step))
```

```
##            median_step
## 2012-10-01          NA
## 2012-10-02           0
## 2012-10-03           0
## 2012-10-04           0
## 2012-10-05           0
## 2012-10-06           0
```

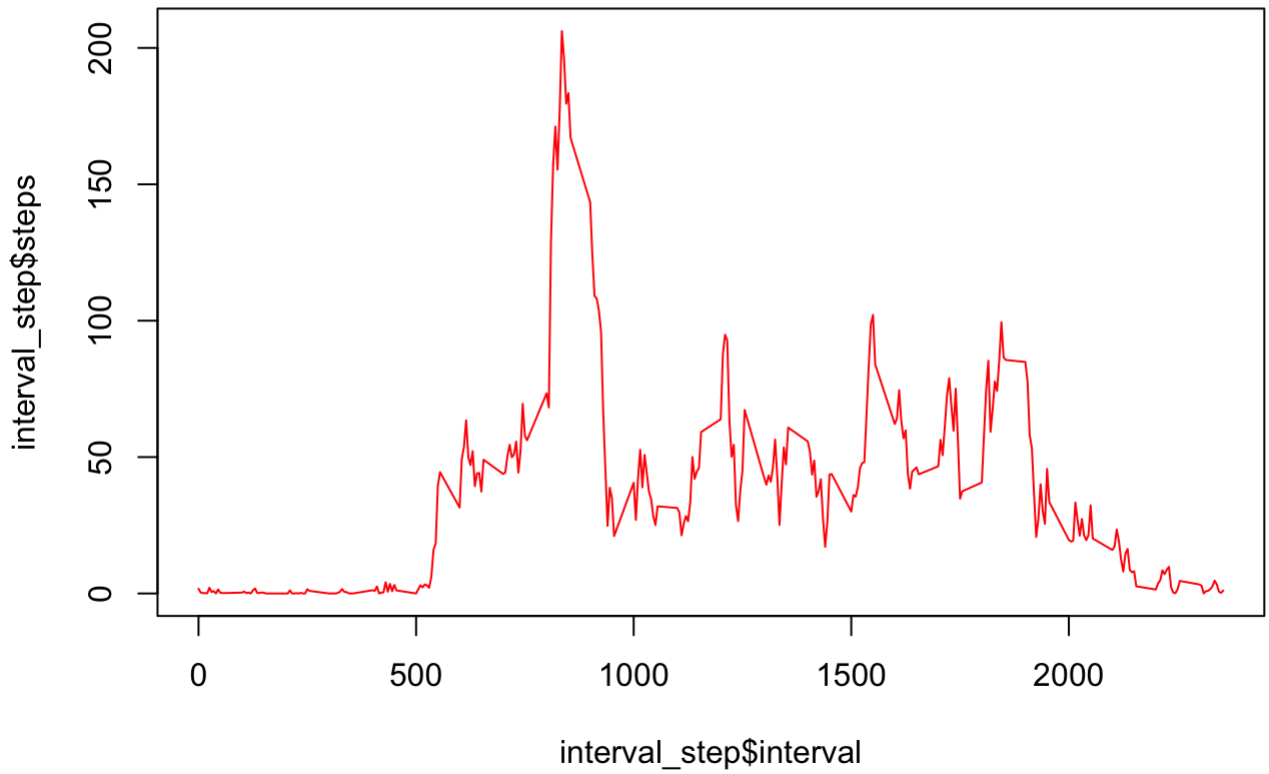
```
head(as.data.frame(mean_step))
```

```
##            mean_step
## 2012-10-01       NaN
## 2012-10-02   0.43750
## 2012-10-03  39.41667
## 2012-10-04  42.06944
## 2012-10-05  46.15972
## 2012-10-06  53.54167
```

2. What is the average daily activity pattern?

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)

```
interval_step<-as.data.frame(tapply(dat$steps,dat$interval,mean,na.rm=T))
interval_step[,2]=rownames(interval_step)
names(interval_step)<-c("steps","interval")
rownames(interval_step)<-1:dim(interval_step)[1]
plot(interval_step$interval,interval_step$steps,type="l",col="red")
```



2 Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with `NA`s)

```
no_NA<-sum(is.na(dat$steps))
no_NA
```

```
## [1] 2304
```

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.

```
dat_new<-dat
for (i in 1:dim(dat_new)[1]) {
  if (is.na(dat_new[i,1])) {
    na.int_value <- dat_new[i,]$interval
    inter_value_steps<-interval_step[interval_step$interval==na.int_value,]$steps
    dat_new[i,1] = inter_value_steps }
}
```

now calculating the median and mean

```
new_median<-tapply(dat_new$steps,dat_new$date,median)
new_mean <- tapply(dat_new$steps,dat_new$date,mean)

as.data.frame(new_median)
```

```
##            new_median
## 2012-10-01   34.11321
## 2012-10-02    0.00000
## 2012-10-03    0.00000
## 2012-10-04    0.00000
## 2012-10-05    0.00000
## 2012-10-06    0.00000
## 2012-10-07    0.00000
## 2012-10-08   34.11321
## 2012-10-09    0.00000
## 2012-10-10    0.00000
## 2012-10-11    0.00000
## 2012-10-12    0.00000
## 2012-10-13    0.00000
## 2012-10-14    0.00000
## 2012-10-15    0.00000
## 2012-10-16    0.00000
## 2012-10-17    0.00000
## 2012-10-18    0.00000
## 2012-10-19    0.00000
## 2012-10-20    0.00000
## 2012-10-21    0.00000
## 2012-10-22    0.00000
## 2012-10-23    0.00000
## 2012-10-24    0.00000
## 2012-10-25    0.00000
## 2012-10-26    0.00000
## 2012-10-27    0.00000
## 2012-10-28    0.00000
## 2012-10-29    0.00000
## 2012-10-30    0.00000
## 2012-10-31    0.00000
## 2012-11-01   34.11321
## 2012-11-02    0.00000
## 2012-11-03    0.00000
## 2012-11-04   34.11321
## 2012-11-05    0.00000
## 2012-11-06    0.00000
## 2012-11-07    0.00000
## 2012-11-08    0.00000
## 2012-11-09   34.11321
## 2012-11-10   34.11321
## 2012-11-11    0.00000
## 2012-11-12    0.00000
## 2012-11-13    0.00000
## 2012-11-14   34.11321
## 2012-11-15    0.00000
## 2012-11-16    0.00000
## 2012-11-17    0.00000
## 2012-11-18    0.00000
## 2012-11-19    0.00000
## 2012-11-20    0.00000
## 2012-11-21    0.00000
## 2012-11-22    0.00000
## 2012-11-23    0.00000
## 2012-11-24    0.00000
## 2012-11-25    0.00000
```

```
## 2012-11-26    0.00000
## 2012-11-27    0.00000
## 2012-11-28    0.00000
## 2012-11-29    0.00000
## 2012-11-30   34.11321
```

```
as.data.frame(new_mean)
```

```
##              new_mean
## 2012-10-01 37.3825996
## 2012-10-02  0.4375000
## 2012-10-03 39.4166667
## 2012-10-04 42.0694444
## 2012-10-05 46.1597222
## 2012-10-06 53.5416667
## 2012-10-07 38.2465278
## 2012-10-08 37.3825996
## 2012-10-09 44.4826389
## 2012-10-10 34.3750000
## 2012-10-11 35.7777778
## 2012-10-12 60.3541667
## 2012-10-13 43.1458333
## 2012-10-14 52.4236111
## 2012-10-15 35.2048611
## 2012-10-16 52.3750000
## 2012-10-17 46.7083333
## 2012-10-18 34.9166667
## 2012-10-19 41.0729167
## 2012-10-20 36.0937500
## 2012-10-21 30.6284722
## 2012-10-22 46.7361111
## 2012-10-23 30.9652778
## 2012-10-24 29.0104167
## 2012-10-25  8.6527778
## 2012-10-26 23.5347222
## 2012-10-27 35.1354167
## 2012-10-28 39.7847222
## 2012-10-29 17.4236111
## 2012-10-30 34.0937500
## 2012-10-31 53.5208333
## 2012-11-01 37.3825996
## 2012-11-02 36.8055556
## 2012-11-03 36.7048611
## 2012-11-04 37.3825996
## 2012-11-05 36.2465278
## 2012-11-06 28.9375000
## 2012-11-07 44.7326389
## 2012-11-08 11.1770833
## 2012-11-09 37.3825996
## 2012-11-10 37.3825996
## 2012-11-11 43.7777778
## 2012-11-12 37.3784722
## 2012-11-13 25.4722222
## 2012-11-14 37.3825996
## 2012-11-15  0.1423611
## 2012-11-16 18.8923611
## 2012-11-17 49.7881944
## 2012-11-18 52.4652778
## 2012-11-19 30.6979167
## 2012-11-20 15.5277778
## 2012-11-21 44.3993056
## 2012-11-22 70.9270833
## 2012-11-23 73.5902778
## 2012-11-24 50.2708333
## 2012-11-25 41.0902778
```

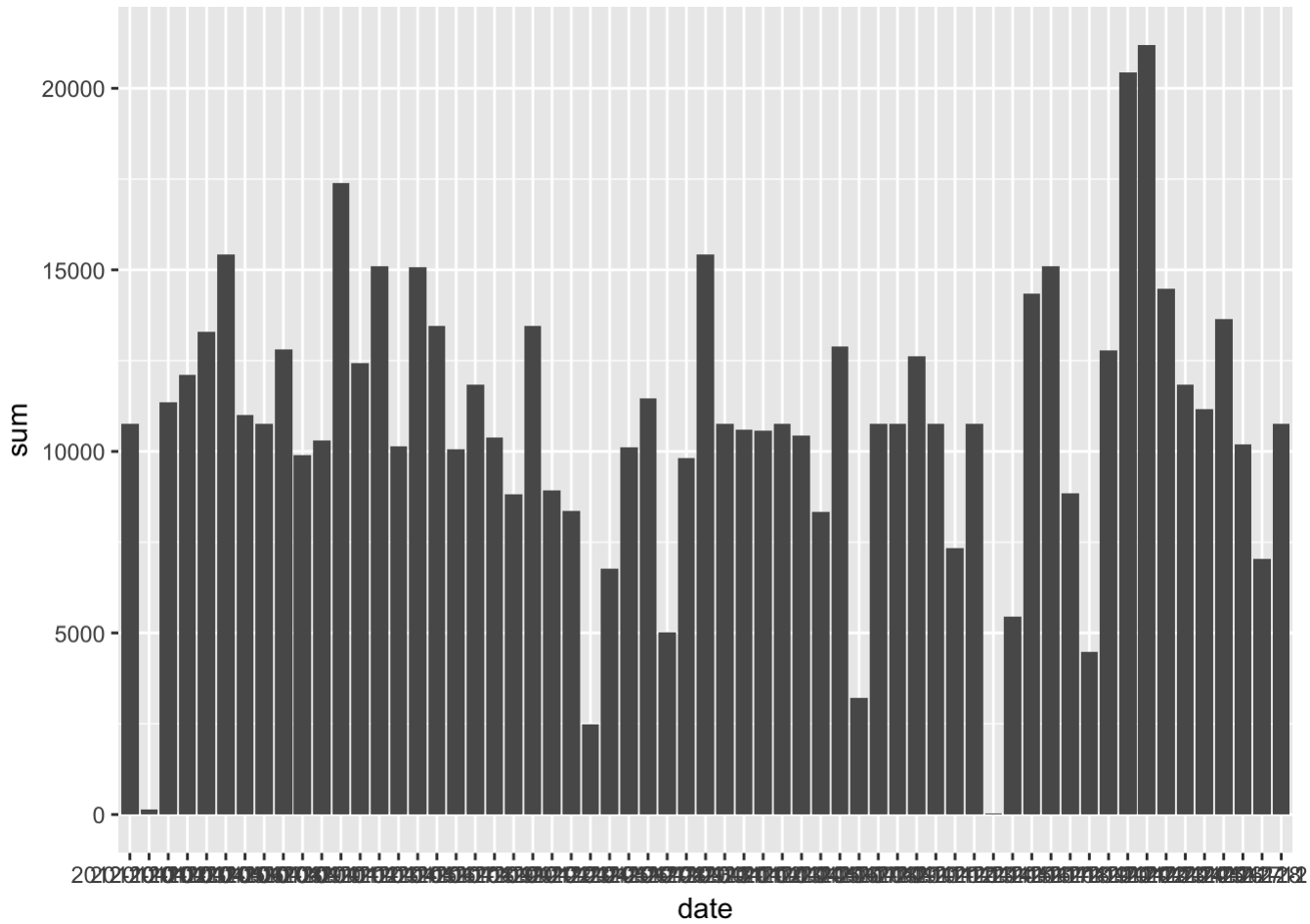
```
## 2012-11-26 38.7569444
## 2012-11-27 47.3819444
## 2012-11-28 35.3576389
## 2012-11-29 24.4687500
## 2012-11-30 37.3825996
```

now calculating the sum of steps per day

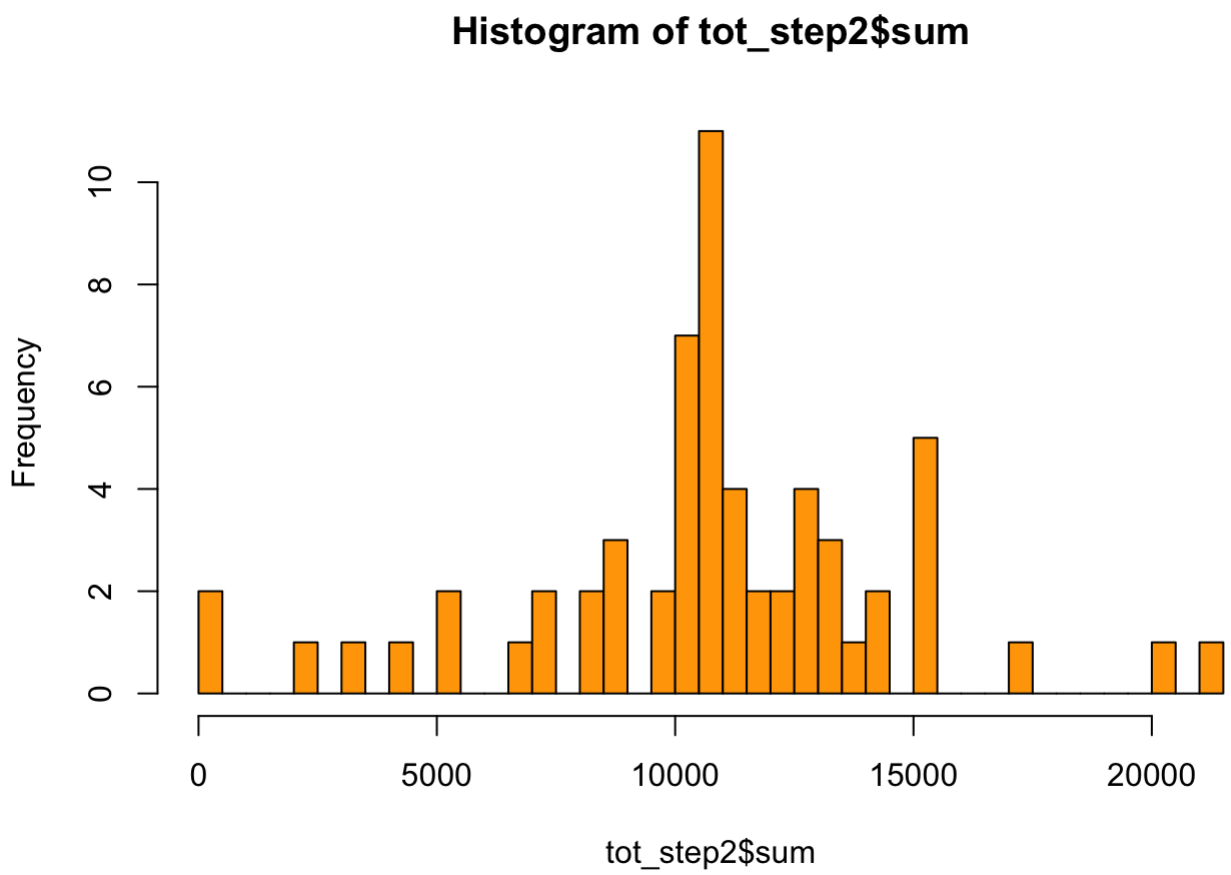
```
tot_step2 <- as.data.frame(tapply(dat_new$steps,dat_new$date,sum))
date2 <- rownames(tot_step2)
rownames(tot_step2)=1:dim(tot_step2)[1]
tot_step2[,2]<-date2
names(tot_step2)<-c("sum","date")
```

This is to plot the bar char and histgram

```
g2<-ggplot(tot_step2,aes(y=sum,x=date))
g2+geom_bar(stat="identity")
```




```
hist(tot_step2$sum,col="orange",breaks=61)
```



Are there differences in activity patterns between weekdays and weekends?

classify workdays and weekends

```
dat_new$weekd<-weekdays(dat_new$date)
for (i in 1:dim(dat_new)[1]) {
   if (dat_new[i,]$weekd %in% c("Monday","Tuesday","Wednesday","Thursday","Friday"))
 {
     dat_new[i,5] <- "workday"
   } else {dat_new[i,5] <- "weekend"
   }
}
names(dat_new)[5]="weekdayclass"
head(dat_new)
```

```
##       steps       date interval  weekd weekdayclass
## 1 1.7169811 2012-10-01        0 Monday      workday
## 2 0.3396226 2012-10-01        5 Monday      workday
## 3 0.1320755 2012-10-01       10 Monday      workday
## 4 0.1509434 2012-10-01       15 Monday      workday
## 5 0.0754717 2012-10-01       20 Monday      workday
## 6 2.0943396 2012-10-01       25 Monday      workday
```

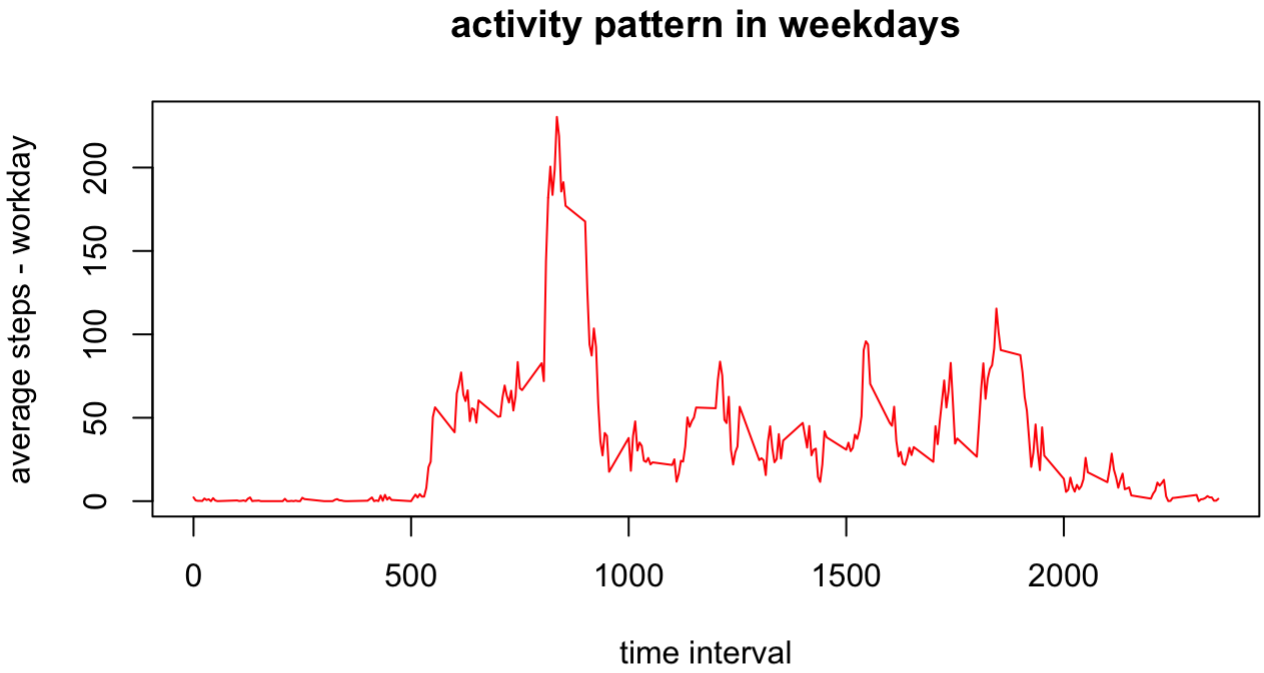
now split the data to weekends and weekdays

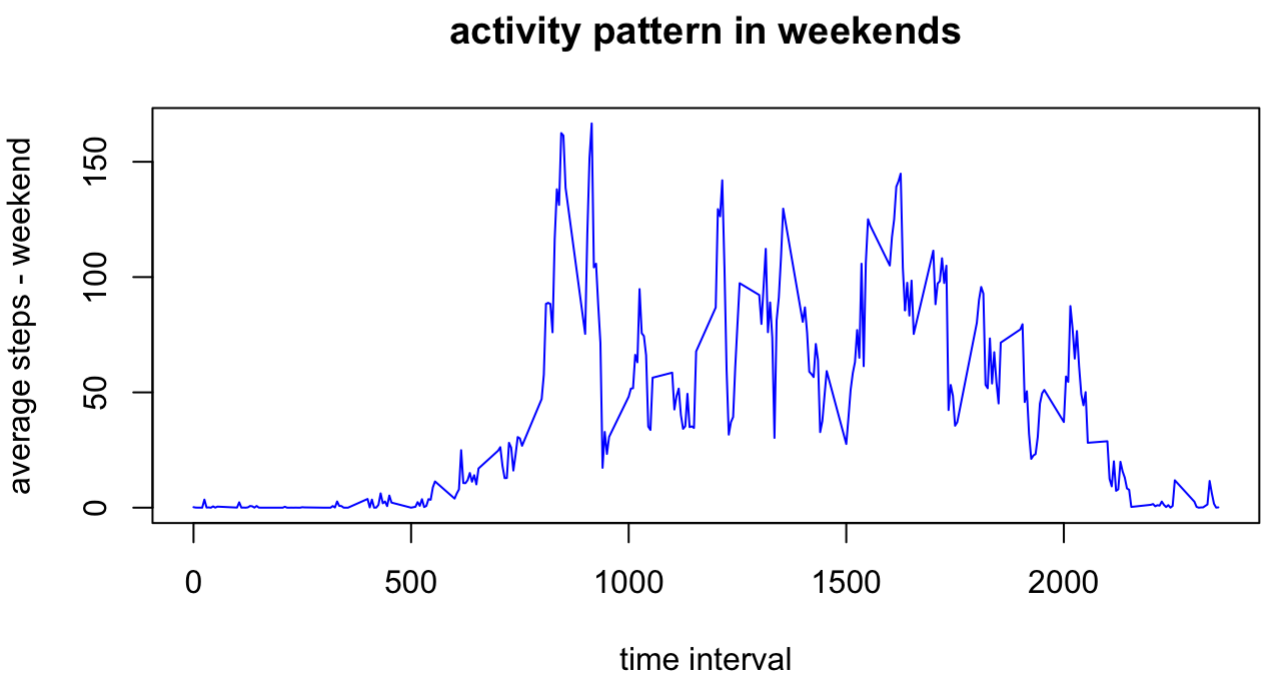
```
dat_new_split<-split.data.frame(dat_new,dat_new$weekdayclass,drop=F)
interv2_weekend<-as.data.frame(tapply(dat_new_split$weekend$steps,dat_new_split$weeke
nd$interval,mean))
interv2_weekend$interval<-rownames(interv2_weekend)
rownames(interv2_weekend)<-1:dim(interv2_weekend)[1]
names(interv2_weekend)<-c("weekend_steps","interval")

interv2_workday<-as.data.frame(tapply(dat_new_split$workday$steps,dat_new_split$workd
ay$interval,mean))
interv2_workday$interval<-rownames(interv2_workday)
rownames(interv2_workday)<-1:dim(interv2_workday)[1]
names(interv2_workday)<-c("workday_steps","interval")
```

plot activity patterns in workdays and weekends

```
par(mfrow=c(2,1))
plot(interv2_workday$interval,interv2_workday$workday_steps,type="l",col="red", ylab=
"average steps - workday", xlab ="time interval", main="activity pattern in weekdays"
)
plot(interv2_weekend$interval,interv2_weekend$weekend_steps,type="l",col="blue", ylab
="average steps - weekend", xlab ="time interval", main="activity pattern in weekend
s")
```


activity pattern in weekdays
average steps - workday
0
50
100
150
200
0
500
1000
1500
2000
time interval


activity pattern in weekends
average steps - weekend
0
50
100
150
0
500
1000
1500
2000
time interval