

peer assignment1

Zuo Yihui

Thursday, July 16, 2015

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#Loading and processing the data
```

```
setwd("E:/Data/reproducible research/repdata-data-activity")
data<-read.csv("activity.csv")
data$date<-as.Date(as.character(data$date),"%Y-%m-%d",tz="UTC")#transform factor to Date
```

What is mean total number of steps taken per day

1 caculate the total number steps taken per day

```
sumsteps_day<-rowsum(data[,c("steps")],group=data$date,na.rm=TRUE,recorder=TRUE)
```

2 histogram of the total number of steps taken each day

```
{r,echo=TRUE}] hist(sumsteps_day,xlab="steps",main="Histogram of steps per day") ## 3 mean
and median of the total number of steps taken per day
```

```
#mean of the total number of steps taken per day
avesteps<-aggregate(x=data[, "steps"],by=list(data$date),FUN=mean)
avesteps
```

```
##      Group.1      x
## 1 2012-10-01      NA
## 2 2012-10-02 0.4375000
## 3 2012-10-03 39.4166667
## 4 2012-10-04 42.0694444
## 5 2012-10-05 46.1597222
## 6 2012-10-06 53.5416667
## 7 2012-10-07 38.2465278
## 8 2012-10-08      NA
## 9 2012-10-09 44.4826389
## 10 2012-10-10 34.3750000
## 11 2012-10-11 35.7777778
## 12 2012-10-12 60.3541667
## 13 2012-10-13 43.1458333
## 14 2012-10-14 52.4236111
## 15 2012-10-15 35.2048611
## 16 2012-10-16 52.3750000
```

```
## 17 2012-10-17 46.7083333
## 18 2012-10-18 34.9166667
## 19 2012-10-19 41.0729167
## 20 2012-10-20 36.0937500
## 21 2012-10-21 30.6284722
## 22 2012-10-22 46.7361111
## 23 2012-10-23 30.9652778
## 24 2012-10-24 29.0104167
## 25 2012-10-25 8.6527778
## 26 2012-10-26 23.5347222
## 27 2012-10-27 35.1354167
## 28 2012-10-28 39.7847222
## 29 2012-10-29 17.4236111
## 30 2012-10-30 34.0937500
## 31 2012-10-31 53.5208333
## 32 2012-11-01 NA
## 33 2012-11-02 36.8055556
## 34 2012-11-03 36.7048611
## 35 2012-11-04 NA
## 36 2012-11-05 36.2465278
## 37 2012-11-06 28.9375000
## 38 2012-11-07 44.7326389
## 39 2012-11-08 11.1770833
## 40 2012-11-09 NA
## 41 2012-11-10 NA
## 42 2012-11-11 43.7777778
## 43 2012-11-12 37.3784722
## 44 2012-11-13 25.4722222
## 45 2012-11-14 NA
## 46 2012-11-15 0.1423611
## 47 2012-11-16 18.8923611
## 48 2012-11-17 49.7881944
## 49 2012-11-18 52.4652778
## 50 2012-11-19 30.6979167
## 51 2012-11-20 15.5277778
## 52 2012-11-21 44.3993056
## 53 2012-11-22 70.9270833
## 54 2012-11-23 73.5902778
## 55 2012-11-24 50.2708333
## 56 2012-11-25 41.0902778
## 57 2012-11-26 38.7569444
## 58 2012-11-27 47.3819444
## 59 2012-11-28 35.3576389
## 60 2012-11-29 24.4687500
## 61 2012-11-30 NA
```

```
#median of the total number of steps taken per day
medsteps<-aggregate(x=data[, "steps"],by=list(data$date),FUN=median)
medsteps
```

```
##      Group.1  x
## 1 2012-10-01 NA
## 2 2012-10-02  0
## 3 2012-10-03  0
```

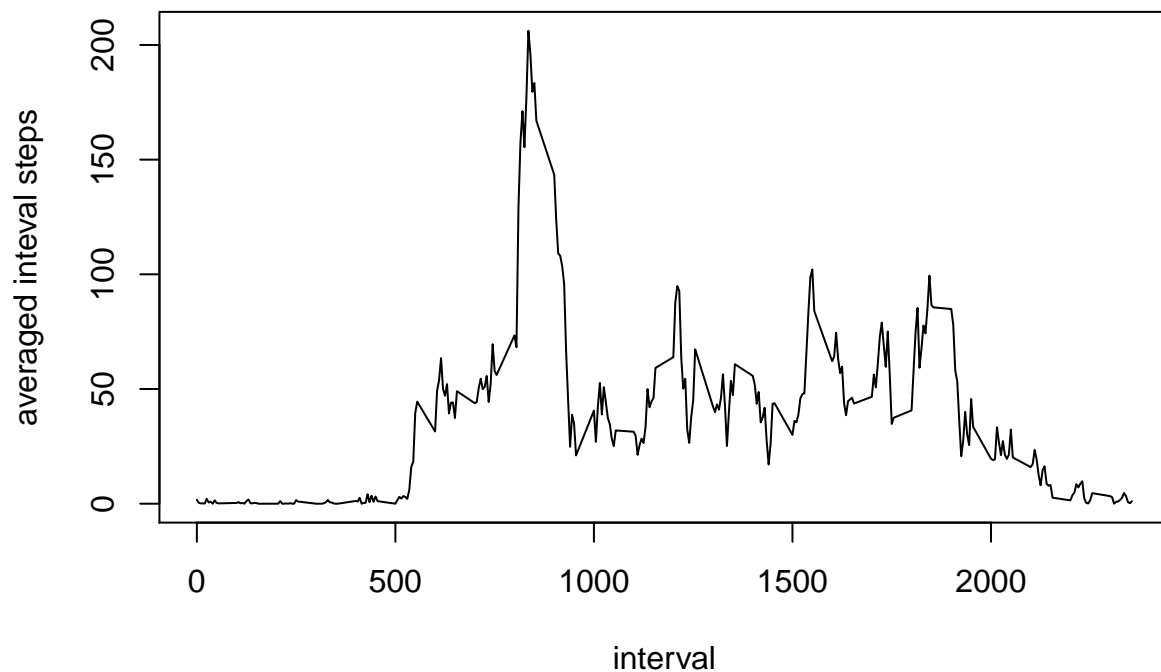
##	4	2012-10-04	0
##	5	2012-10-05	0
##	6	2012-10-06	0
##	7	2012-10-07	0
##	8	2012-10-08	NA
##	9	2012-10-09	0
##	10	2012-10-10	0
##	11	2012-10-11	0
##	12	2012-10-12	0
##	13	2012-10-13	0
##	14	2012-10-14	0
##	15	2012-10-15	0
##	16	2012-10-16	0
##	17	2012-10-17	0
##	18	2012-10-18	0
##	19	2012-10-19	0
##	20	2012-10-20	0
##	21	2012-10-21	0
##	22	2012-10-22	0
##	23	2012-10-23	0
##	24	2012-10-24	0
##	25	2012-10-25	0
##	26	2012-10-26	0
##	27	2012-10-27	0
##	28	2012-10-28	0
##	29	2012-10-29	0
##	30	2012-10-30	0
##	31	2012-10-31	0
##	32	2012-11-01	NA
##	33	2012-11-02	0
##	34	2012-11-03	0
##	35	2012-11-04	NA
##	36	2012-11-05	0
##	37	2012-11-06	0
##	38	2012-11-07	0
##	39	2012-11-08	0
##	40	2012-11-09	NA
##	41	2012-11-10	NA
##	42	2012-11-11	0
##	43	2012-11-12	0
##	44	2012-11-13	0
##	45	2012-11-14	NA
##	46	2012-11-15	0
##	47	2012-11-16	0
##	48	2012-11-17	0
##	49	2012-11-18	0
##	50	2012-11-19	0
##	51	2012-11-20	0
##	52	2012-11-21	0
##	53	2012-11-22	0
##	54	2012-11-23	0
##	55	2012-11-24	0
##	56	2012-11-25	0
##	57	2012-11-26	0

```
## 58 2012-11-27 0
## 59 2012-11-28 0
## 60 2012-11-29 0
## 61 2012-11-30 NA
```

What is the average daily activity pattern?

1. Make a time series plot of the 5-minute interval (x-axis) and the average number of steps taken, averaged across all days (y-axis)

```
library(reshape2)
data_m<-melt(data,id=c("interval","date"),na.rm=TRUE)
ave_intersteps<-acast(data_m,interval~variable,mean)
plot(x=rownames(ave_intersteps),y=ave_intersteps,type="l",xlab="interval",ylab="averaged interval steps")
```



2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps? `rownames(ave_intersteps)[which.max(ave_intersteps)]` “ “ ##imputing missing value ###
1. Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs

```
n<-length(data$steps[is.na(data$steps)])
n
```

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

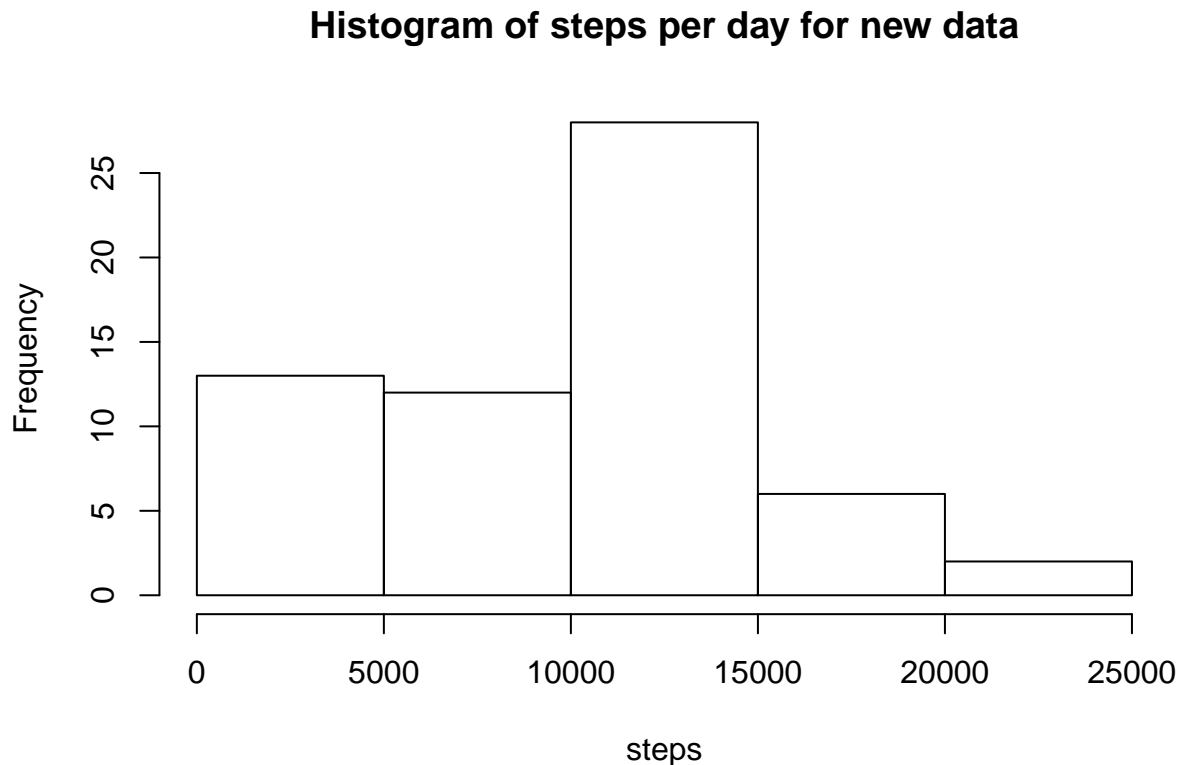
2-3. fill the missing value with mean of 5-minute interval

use the mean of 5-minutes interval to fill the missing value

```
Newdata<-data
Newdata$steps[is.na(data$steps)]<-ave_intersteps[is.na(data$steps)]
```

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.

```
#total number of steps taken every day
New_sumsteps_day<-rowsum(data[,c("steps")],group=Newdata$date,na.rm=TRUE,recorder=TRUE)
hist(New_sumsteps_day,xlab="steps",main="Histogram of steps per day for new data")
```



```
#mean of the total number of steps taken per day
new_avesteps<-aggregate(x=Newdata[, "steps"],by=list(Newdata$date),FUN=mean)

#median of the total number of steps taken per day
new_medsteps<-aggregate(x=Newdata[, "steps"],by=list(Newdata$date),FUN=median)
```

Are there differences in activity patterns between weekdays and weekends?

```
library(timeDate)
```

```
## Warning: package 'timeDate' was built under R version 3.1.3
```

```
library(ggplot2)
```

```
library(grid)
```

```
#1.Create a new factor variable in the dataset with two levels - "weekday" and "weekend"
```

```
Newdata$timeDate[isWeekday(Newdata$date)]<-"weekday"
```

```
Newdata$timeDate[!isWeekday(Newdata$date)]<-"weekend"
```

```
#2.Make a panel plot containing a time series plot (i.e. type = "l") of the 5-minute interval (x-axis)
```

```
ave_weeksteps<-aggregate(x=Newdata[, "steps"],by=list(Newdata$timeDate,Newdata$interval),FUN=mean,na.rm=
```

```
colnames(ave_weeksteps)<-c("weekdays", "interval", "steps")
```

```
p<-ggplot(data=ave_weeksteps,aes(y=steps,x=interval))+geom_line(colour="blue")+facet_wrap(~weekdays,ncol=
```

```
p+ylab("Number of steps")+theme(strip.background=element_rect(fill="lightblue"))
```

