Week 2 assignment

run in the directory of “activity.csv”

Load packages

```
library(dplyr); library(ggplot2);library(dplyr); library(ggplot2)
```

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

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

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

```
library(lubridate)
```

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

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

Read data

```
data<-read.csv("activity.csv", header=T)
data$date<-as.Date(data$date)
```

The total number of steps taken each day with

```
N.Steps.Day<-data %>% group_by(date) %>% summarise(Number.Steps= sum(steps)) %>%
    data.frame
g<- ggplot(N.Steps.Day, aes(Number.Steps)) + geom_histogram()+
    xlab("Total Number of Steps per day") + ylab("Frequency")+
    ggtitle("Total number of steps taken each day")
cat("The mean and median number of steps taken per day")
```

```
## The mean and median number of steps taken per day
```

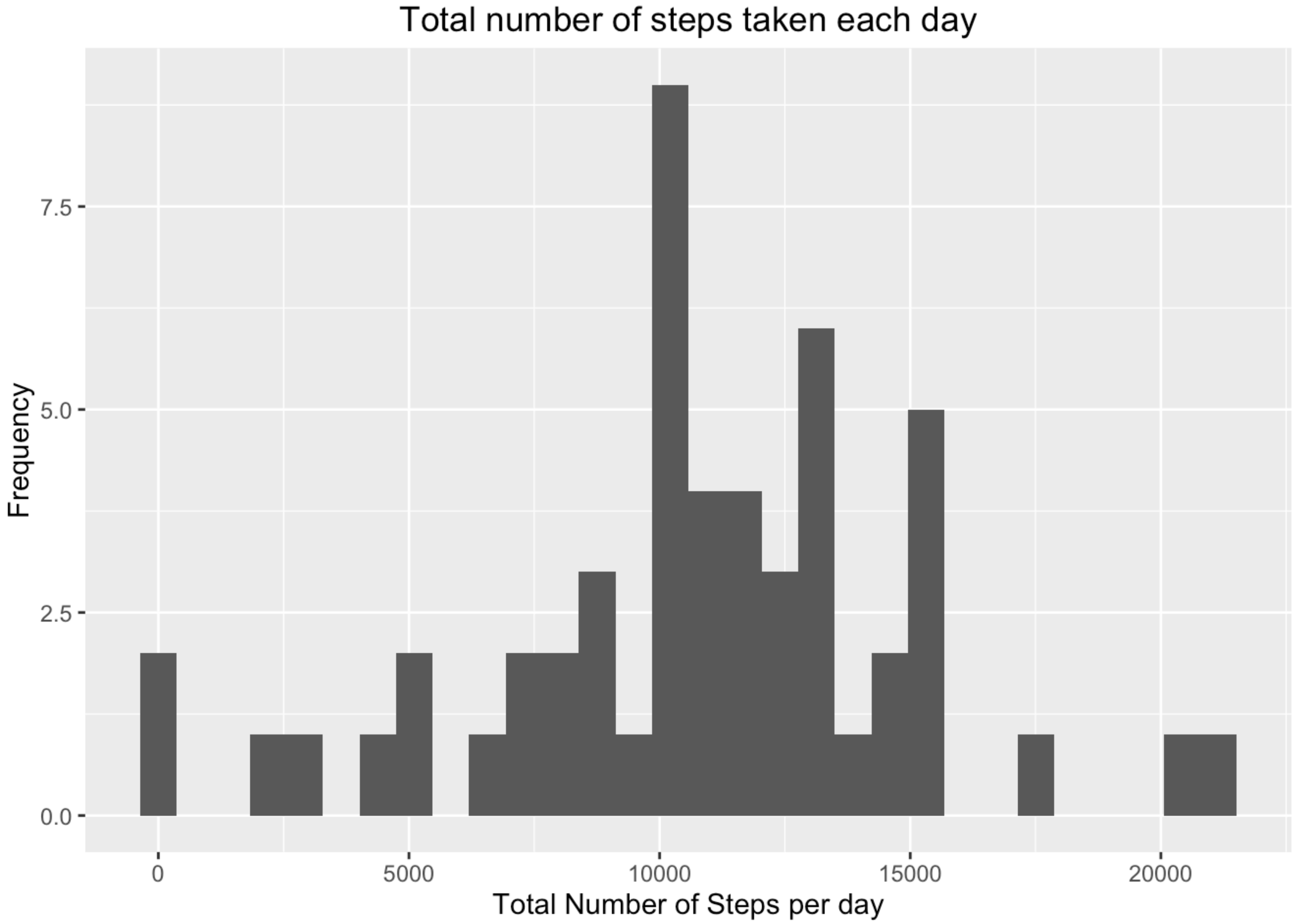
```
N.Steps.Day %>% summarise(Mean_number_steps=mean(Number.Steps, na.rm=T),
                          Medain_number_steps=median(N.Steps.Day$Number.Steps, na.rm=
T))
```

```
##   Mean_number_steps Medain_number_steps
## 1          10766.19               10765
```

```
print(g)
```

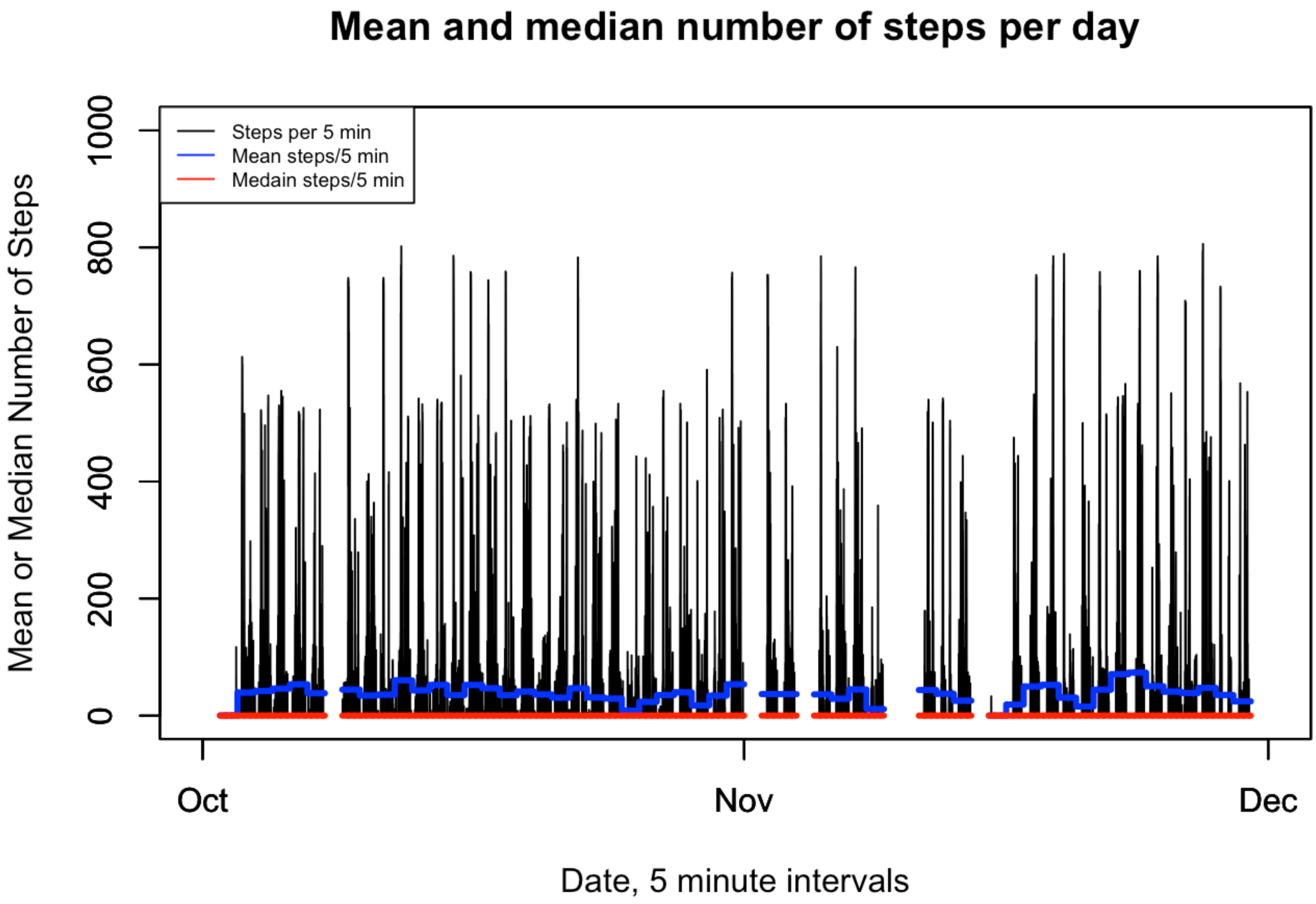
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



##What is the average daily activity pattern?

```
Day.Mean.Median<-data %>% group_by(date) %>% select(steps:date) %>%
        summarise(mean= mean(steps), median=median(steps)) %>% data.frame
data$mean <- NA; data$median <- NA
Dates<-unique(data$date)
for(date in Dates){
    data$mean[data$date == date] <-
        Day.Mean.Median$mean[Day.Mean.Median$date ==date]
    data$median[data$date == date] <-
        Day.Mean.Median$median[Day.Mean.Median$date ==date]
}
data$hour <-data$interval %/% 100
data$minute <-data$interval %% 100
data$date2<-ymd(data$date)
hour(data$date2) <-data$hour
minute(data$date2) <-data$minute
with(data, plot(date2, steps, type="l", col='black',
                ylim = c(0,1000), ylab="", xlab=""))
par(new=T)
with(data, plot(date2, mean, type="l", col='blue', pch=20,lwd=3,
    ylim = c(0,1000), ylab="", xlab=""))
par(new=T)
with(data, plot(date2, median, type="l", col='red', pch=18,lwd=3,
    ylim = c(0,1000), ylab="Mean or Median Number of Steps",
    xlab="Date, 5 minute intervals",
    main="Mean and median number of steps per day"))
legend("topleft", legend=c("Steps per 5 min", "Mean steps/5 min","Medain steps/5 min"
),
       lty = 1, col=c("black","blue","red"), cex=0.6)
```



##Which 5-minute interval contains Max?

```
max.steps<- tapply(data$steps, data$date, max)
library(foreach)
max_step_day_interval <- foreach(i =1:length(max.steps), .combine = rbind)%do% {
      with(data, data[!is.na(steps) & date== names(max.steps)[i] &
                    steps==max.steps[[2]], c(2,3)])
}
print(max_step_day_interval)
```

```
##              date interval
## 555   2012-10-02     2210
## 1981  2012-10-07     2100
## 3259  2012-10-12      730
## 3260  2012-10-12      735
## 5899  2012-10-21     1130
## 6828  2012-10-24     1655
## 8170  2012-10-29      845
## 9304  2012-11-02      715
## 9719  2012-11-03     1750
## 16069 2012-11-25     1900
```

Imputing missing values

*Number of rows with missing “steps” values

```
sum(is.na(data$steps))
```

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

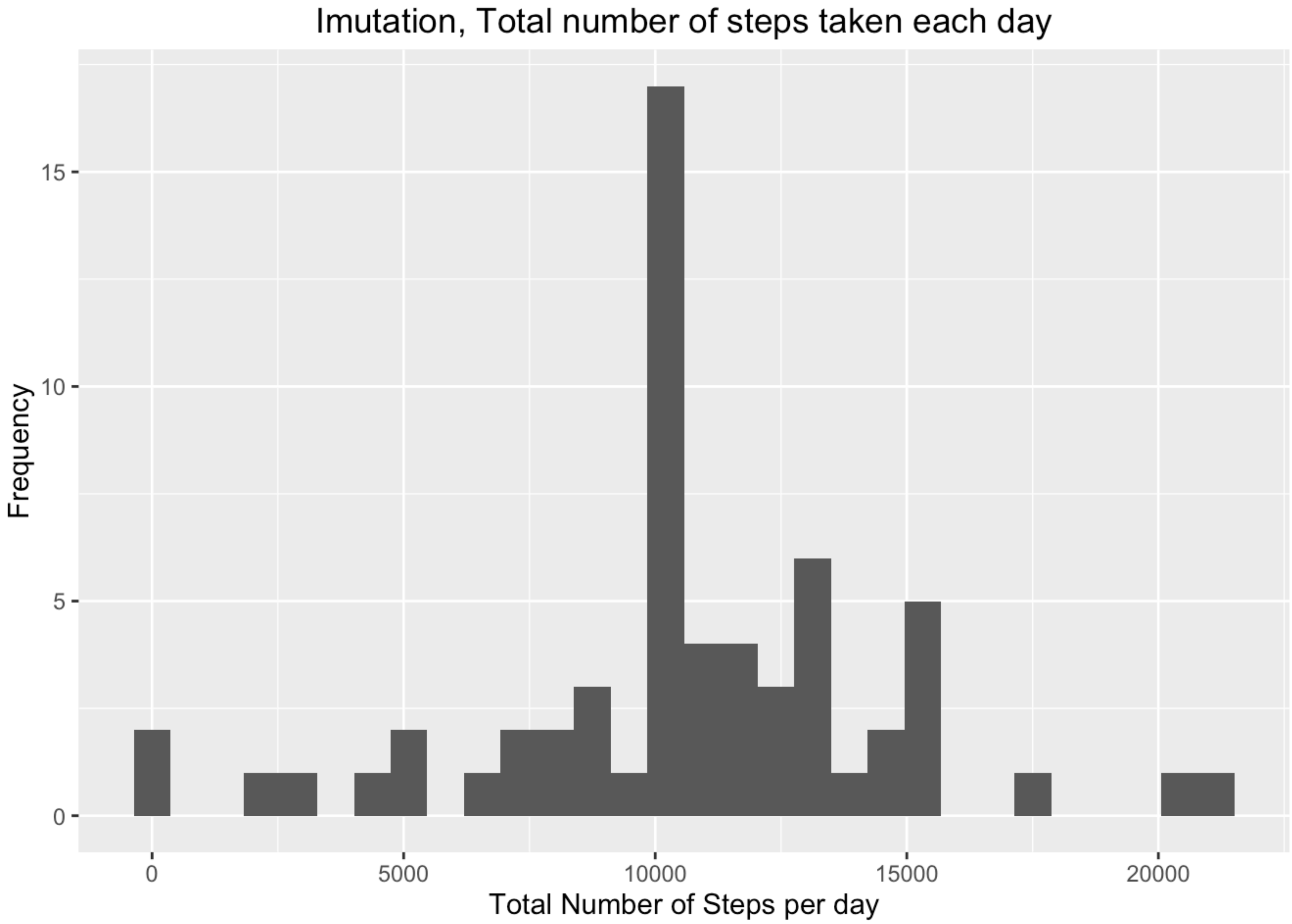
*Fill these missing cells with value from 0 to overall mean + 30% sd

```
m<-mean(data$steps,na.rm=T); sd<-sd(data$steps,na.rm=T)
set.seed(123)
imputation=runif(sum(is.na(data$steps)), 0, m+0.3*sd)
data$steps[is.na(data$steps)]<- imputation
```

The total number of steps taken each day with imputation

```
N.Steps.Day<-data %>% select(steps:interval) %>% group_by(date) %>% summarise(Number.Steps= sum(steps)) %>% data.frame
ggplot(N.Steps.Day, aes(Number.Steps)) + geom_histogram()+
    xlab("Total Number of Steps per day") + ylab("Frequency")+
    ggtitle("Imutation, Total number of steps taken each day")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



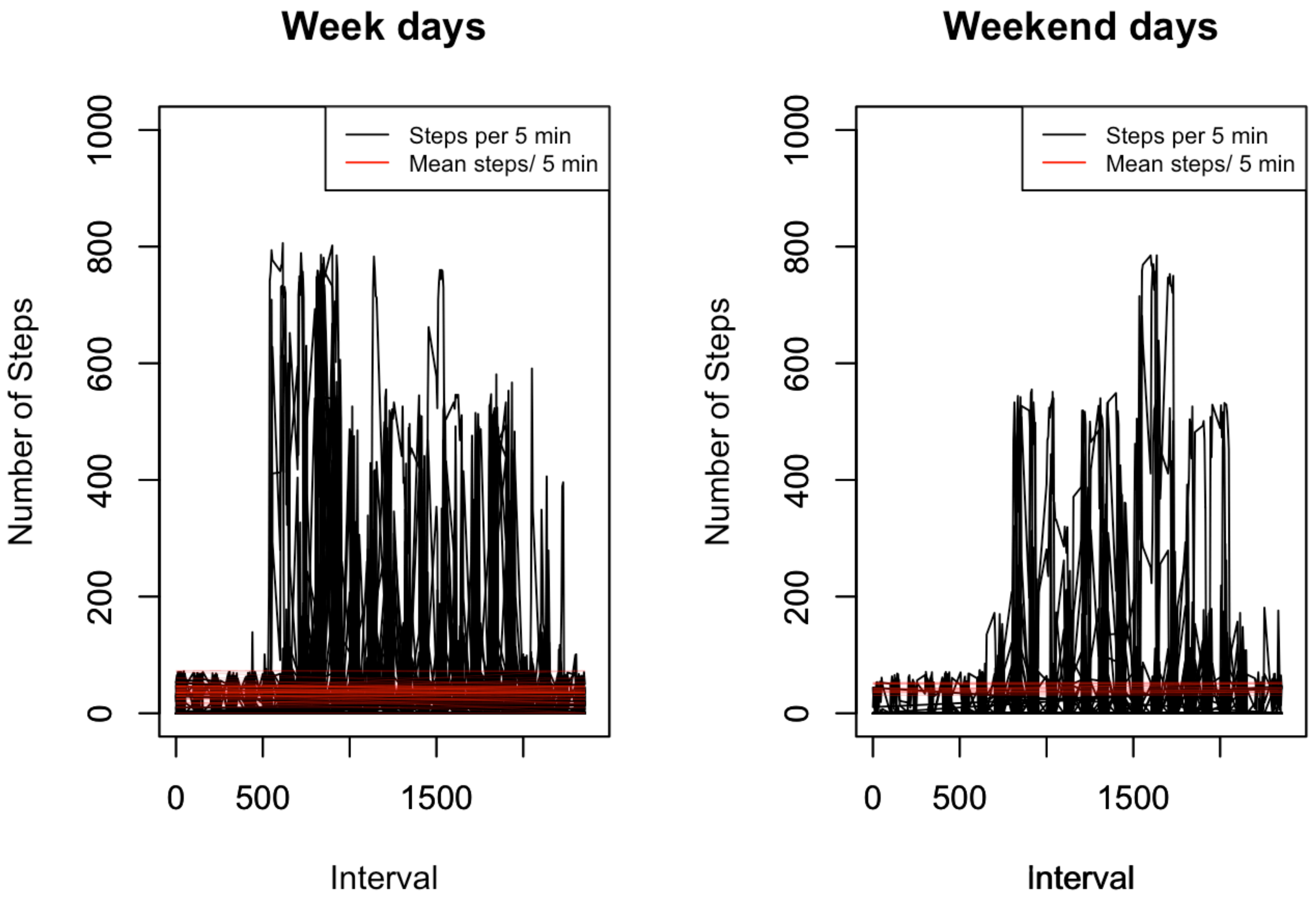
##The mean and median total number of steps taken per day with imputation

```
N.Steps.Day %>% summarise(Mean_number_steps=mean(Number.Steps, na.rm=T),
                          Medain_number_steps=median(N.Steps.Day$Number.Steps, na.rm=
T))
```

```
##   Mean_number_steps Medain_number_steps
## 1          10688.61               10395
```

Are there differences in activity patterns between weekdays and weekends?

```
data$weekday <- weekdays(data$date)
data$weekday[data$weekday %in% c("Sunday", "Saturday")] <- "weekend"
data$weekday[data$weekday %in% c("Monday", "Tuesday", "Wednesday","Thursday","Friday"
)] <- "weekday"
#split into weekday and weekend data
weekday<-data[data$weekday =="weekday",]
weekend<-data[data$weekday =="weekend",]
#plot weekday and weekend days
par(mfrow=c(1,2))
#plot week day
with(weekday, plot(interval, steps, type="l", col='black', xlim = c(0,2400),
                ylim = c(0,1000), ylab="", xlab=""))
par(new=T)
with(weekday, plot(interval, mean, type="l", col='red', lwd=0.1, xlim = c(0,2400),
    ylim = c(0,1000), ylab="Number of Steps", xlab="Interval"))
legend("topright", legend=c("Steps per 5 min", "Mean steps/ 5 min"),
       lty = 1, col=c("black","red"), cex = 0.7)
title(main = "Week days")
#plot weekend day
with(weekend, plot(interval, steps, type="l", col='black',xlim = c(0,2400),
                ylim = c(0,1000), ylab=""))
par(new=T)
with(weekend, plot(interval, mean, type="l", col='red', lwd=0.1, xlim = c(0,2400),
    ylim = c(0,1000), ylab="Number of Steps", xlab="Interval"))
legend("topright", legend=c("Steps per 5 min", "Mean steps/ 5 min"),
       lty = 1, col=c("black","red"), cex = 0.7)
title(main = "Weekend days")
```


Week days
Weekend days
Steps per 5 min
Mean steps/ 5 min
Number of Steps
0
200
400
600
800
1000
0
500
1500
Interval