Reproducible Research Course Project 1

Thomas Zachariah

6/9/2017

1. Code for Loading and examining the data

The data is downloaed into the current working directory

```
setwd("~/Desktop/5ReproducibleResearch")
suppressMessages(library(plyr))
suppressMessages(library(dplyr))
#name of data file: pamdd (personal activity monitoring device data)
pamdd <- read.csv("activity.csv", header = TRUE, sep = ",")
names(pamdd)
```

```
## [1] "steps"    "date"     "interval"
```

```
dim(pamdd)
```

```
## [1] 17568     3
```

```
head(pamdd)
```

```
##   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
```

```
str(pamdd)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : Factor w/ 61 levels "2012-10-01","2012-10-02",..: 1 1 1 1 1 1 1 1 1 1
...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

```
summary(pamdd)
```

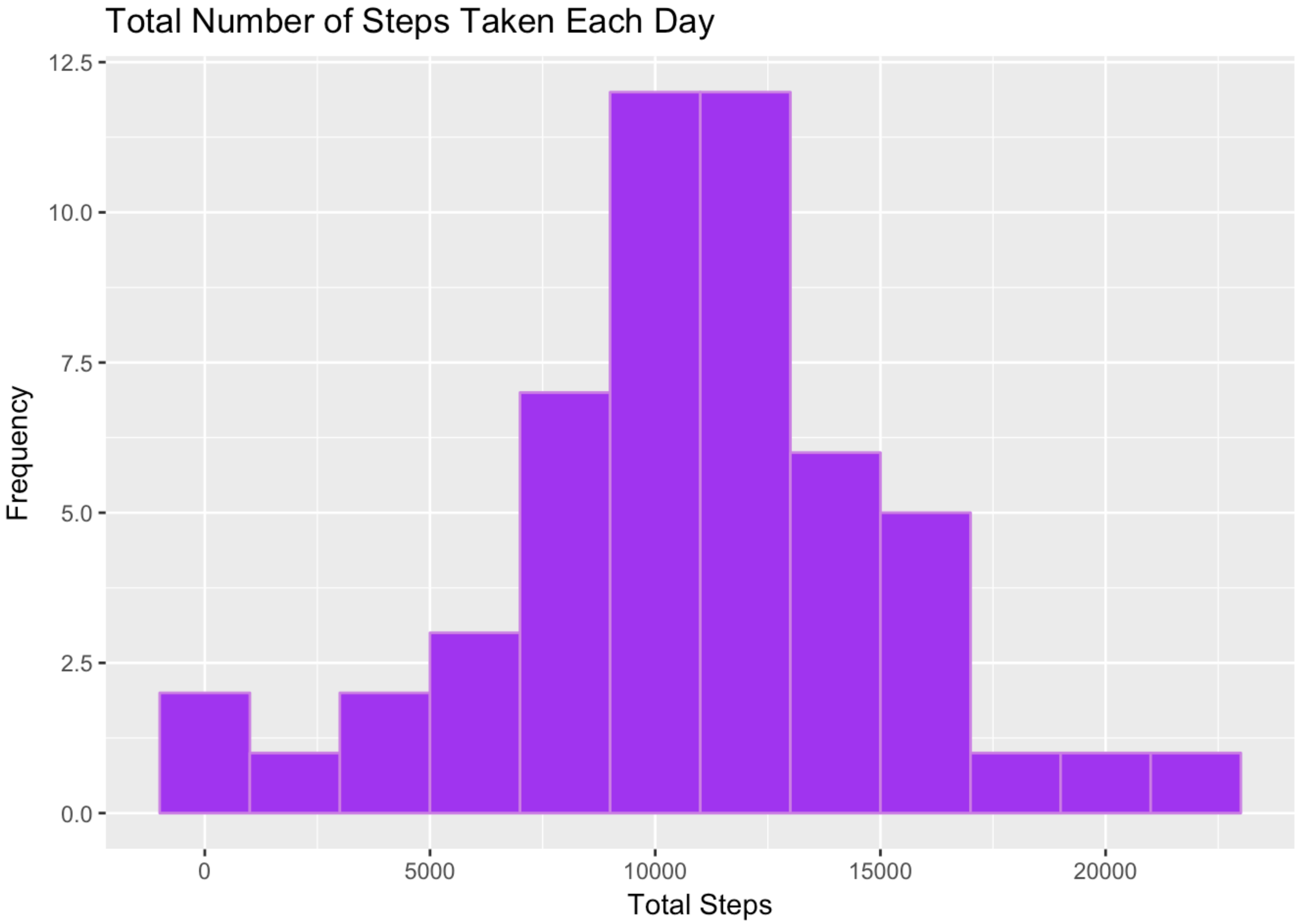
```
##      steps                date          interval     
##  Min.   :  0.00   2012-10-01:  288   Min.   :   0.0  
##  1st Qu.:  0.00   2012-10-02:  288   1st Qu.: 588.8  
##  Median :  0.00   2012-10-03:  288   Median :1177.5  
##  Mean   : 37.38   2012-10-04:  288   Mean   :1177.5  
##  3rd Qu.: 12.00   2012-10-05:  288   3rd Qu.:1766.2  
##  Max.   :806.00   2012-10-06:  288   Max.   :2355.0  
##  NA's   :2304     (Other)   :15840                   
```

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

```
#calculating the number of steps taken
steps <- pamdd %>% filter(!is.na(steps)) %>% group_by(date) %>% summarize(steps = sum
(steps)) %>% print
```

```
## # A tibble: 53 x 2
##          date steps
##        <fctr> <int>
##  1 2012-10-02   126
##  2 2012-10-03 11352
##  3 2012-10-04 12116
##  4 2012-10-05 13294
##  5 2012-10-06 15420
##  6 2012-10-07 11015
##  7 2012-10-09 12811
##  8 2012-10-10  9900
##  9 2012-10-11 10304
## 10 2012-10-12 17382
## # ... with 43 more rows
```

```
#drawigh the histogram
library(ggplot2)
ggplot(steps, aes(x = steps)) +
geom_histogram(fill = "purple", color=rgb(.8,.5,.9), binwidth = 2000) +
labs(title = "Total Number of Steps Taken Each Day", x = "Total Steps", y = "Frequenc
y")
```



3. Mean and median number of steps taken each day

```
paste("mean number of steps =", mean(steps$steps, na.rm = TRUE))
```

```
## [1] "mean number of steps = 10766.1886792453"
```

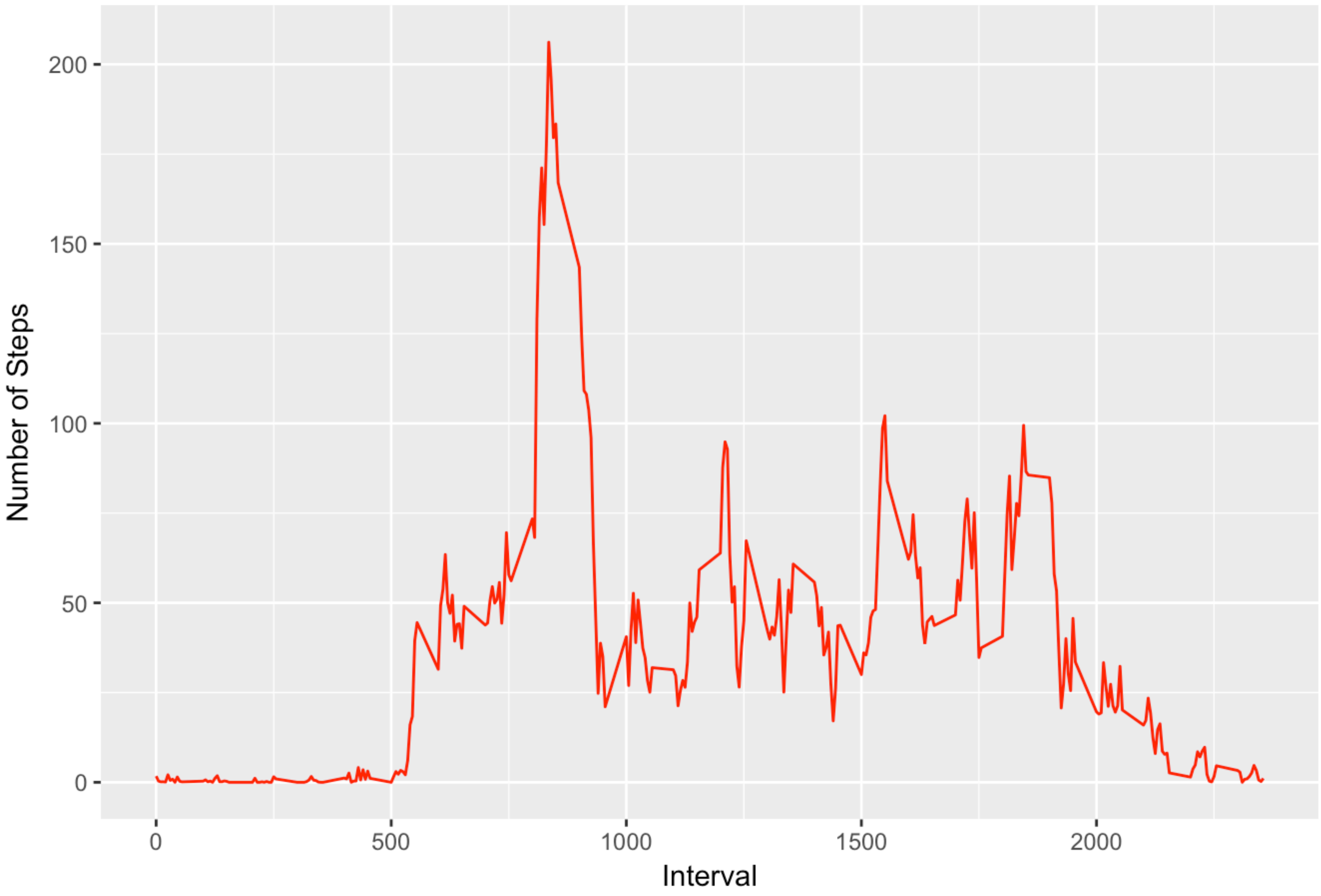
```
paste("median number of steps = ", median(steps$steps, na.rm = TRUE))
```

```
## [1] "median number of steps =  10765"
```

4. Time series plot of the average number of steps taken

```
TSdata <- pamdd %>% filter(!is.na(steps)) %>% group_by(interval) %>% summarize(steps = mean(steps))
ggplot(TSdata, aes(x=interval, y=steps)) + geom_line(color = "red") + labs(title = "Time series plot of the average number of steps taken", x = "Interval", y = "Number of Steps")
```

Time series plot of the average number of steps taken



###5. The 5-minute interval that, on average, contains the maximum number of steps

```
TSdata[which.max(TSdata$steps),]
```

```
## # A tibble: 1 x 2
##   interval    steps
##      <int>    <dbl>
## 1      835 206.1698
```

6. Code to describe and show a strategy for imputing missing data

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

ii. 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.

iii. Create a new dataset that is equal to the original dataset but with the missing data filled in.

iv. 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. 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?

```
# (i)
sum(is.na(pamdd$steps))
```

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

ii. Creating a new data set by replacing “NA” by the average number of steps in the same 5-min interval

```
new_data <- pamdd
nas <- is.na(new_data$steps)
avg_interval <- tapply(new_data$steps,new_data$interval, mean, na.rm=TRUE, simplify=T
RUE)
new_data$steps[nas] <- avg_interval[as.character(new_data$interval[nas])]
#Checking the missing values
sum(is.na(new_data$steps))
```

```
## [1] 0
```

```
new_pamdd <- na.omit(pamdd) # original data with "NA" omitted
dim(new_pamdd); head(new_pamdd)
```

```
## [1] 15264     3
```

```
##     steps       date interval
## 289     0 2012-10-02        0
## 290     0 2012-10-02        5
## 291     0 2012-10-02       10
## 292     0 2012-10-02       15
## 293     0 2012-10-02       20
## 294     0 2012-10-02       25
```

```
dim(new_data); head(new_data) # original data with "NA" replaced
```

```
## [1] 17568     3
```

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

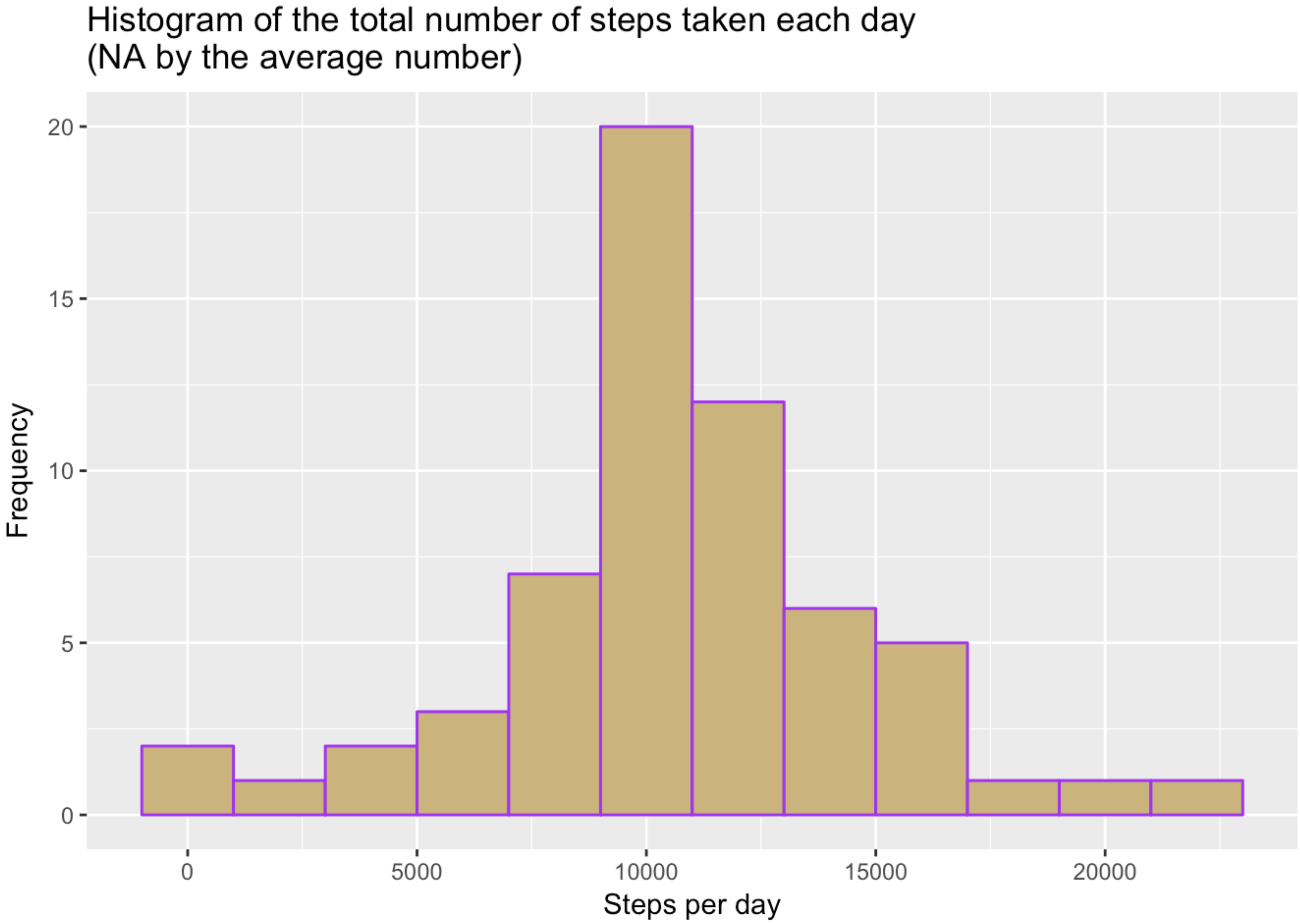
iii. Calculating the number of steps taken in each 5-minute interval per day

```
new_steps <- new_data %>%
  filter(!is.na(steps)) %>%
  group_by(date) %>%
  summarize(steps = sum(steps)) %>%
  print
```

```
## # A tibble: 61 x 2
##          date    steps
##        <fctr>    <dbl>
##  1 2012-10-01 10766.19
##  2 2012-10-02   126.00
##  3 2012-10-03 11352.00
##  4 2012-10-04 12116.00
##  5 2012-10-05 13294.00
##  6 2012-10-06 15420.00
##  7 2012-10-07 11015.00
##  8 2012-10-08 10766.19
##  9 2012-10-09 12811.00
## 10 2012-10-10  9900.00
## # ... with 51 more rows
```

iv. Make a histogram, calculate mean, median, etc.

```
ggplot(new_steps, aes(x = steps)) +
  geom_histogram(fill = rgb(.8,.7,.5),color="purple", binwidth = 2000) +
  labs(title = "Histogram of the total number of steps taken each day\n(NA by the ave
rage number)", x = "Steps per day", y = "Frequency")
```



```
mean_new_steps <- mean(new_steps$steps, na.rm = TRUE); print("mean of new_steps =");
mean_new_steps
```

```
## [1] "mean of new_steps ="
```

```
## [1] 10766.19
```

```
median_new_steps <- median(new_steps$steps, na.rm = TRUE); print("median of new_steps
="); median_new_steps
```

```
## [1] "median of new_steps ="
```

```
## [1] 10766.19
```

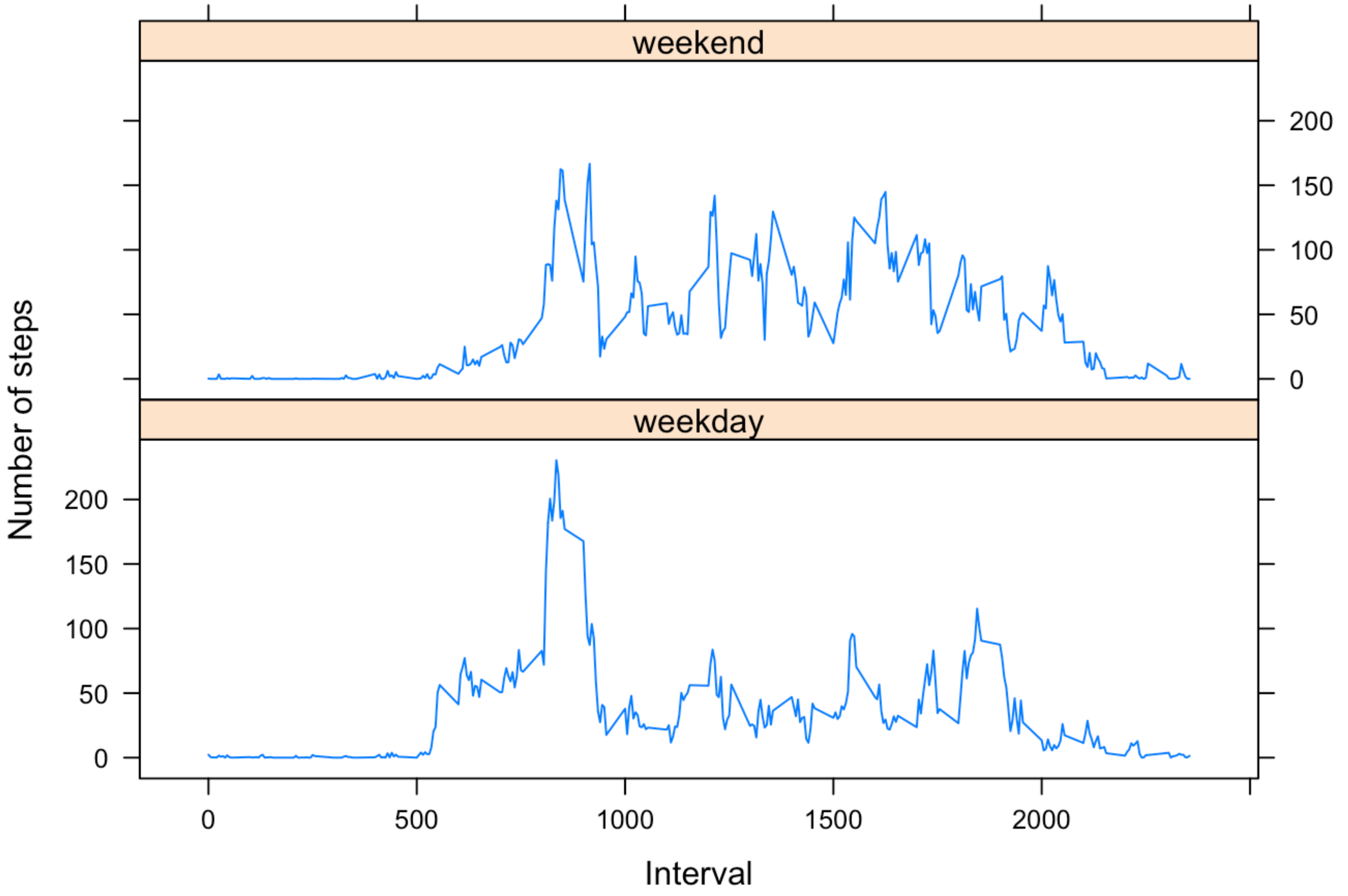
Are there differences in activity patterns between weekdays and weekends?

Making a panel plot containing 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 weekday days or weekend days (y-axis)

```
suppressMessages(library(Hmisc))
Sys.setlocale("LC_TIME", locale = "")
```

```
## [1] "en_US.UTF-8"
```

```
new_data$weekdays <- weekdays(as.Date(new_data$date))
new_data$weekdays <- ifelse(new_data$weekdays %in% c("Saturday", "Sunday"),"weekend",
"weekday")
average <- ddply(new_data, .(interval, weekdays), summarise, steps=mean(steps))
#creating the plot
library(lattice)
xyplot(steps ~ interval | weekdays, data = average, layout = c(1, 2), type="l", xlab
= "Interval", ylab = "Number of steps")
```



library(lattice) xyplot(steps ~ interval | weekdays, data = average, layout = c(1, 2), type="l", xlab = "Interval", ylab = "Number of steps")