Reproducible Research: Peer Assessment 1

Loading and processing the data

```
library(ggplot2)
unzip("activity.zip")
```

```
## Warning in unzip("activity.zip"): error 1 in extracting from zip file
```

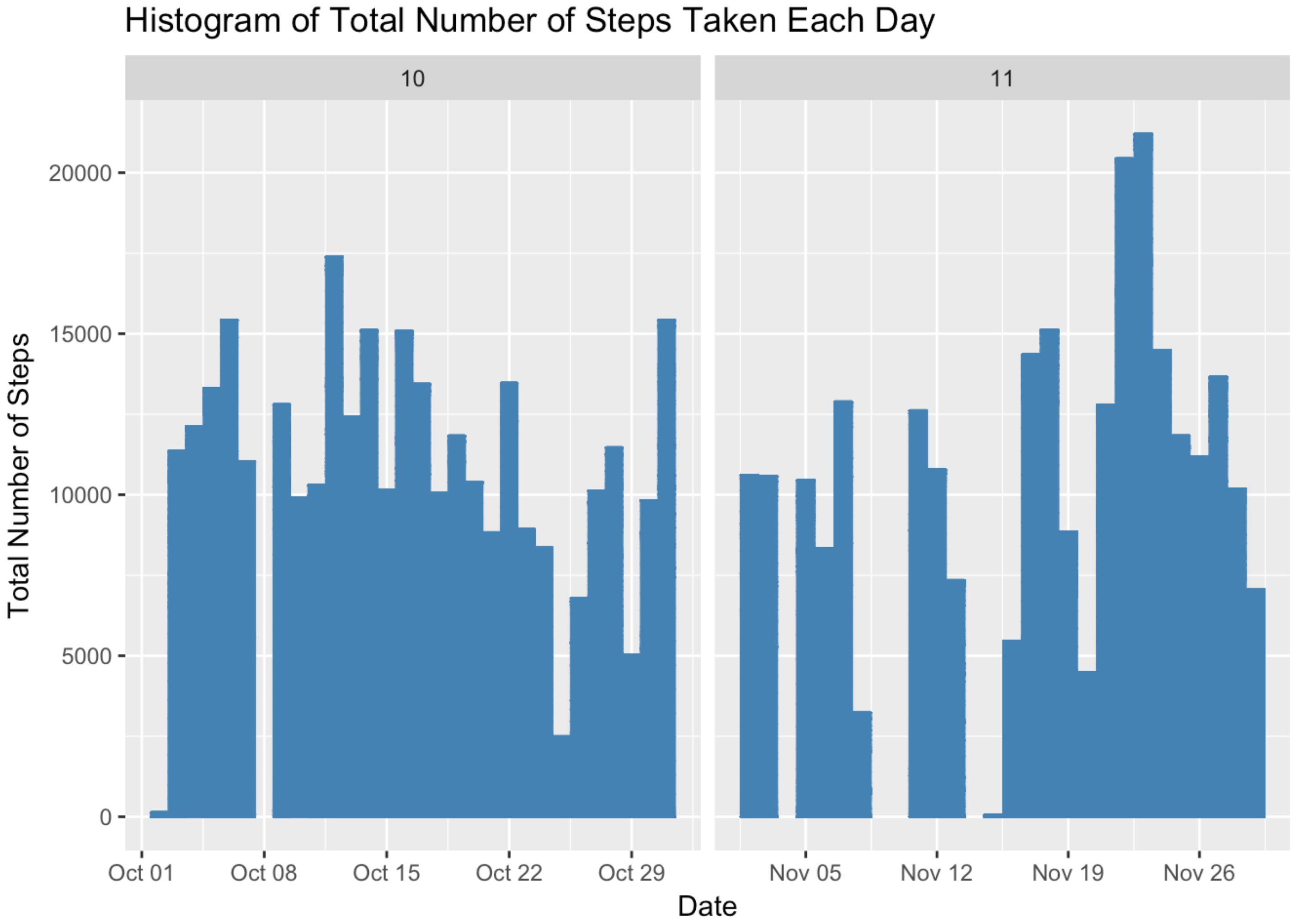
```
data <- read.csv("activity.csv", colClasses = c("integer", "Date", "factor"))
data$month <- as.numeric(format(data$date, "%m"))
```

Calculate the total/mean/median number of steps taken per day

```
StepsByday <- tapply(data$steps, data$date, sum, na.rm = TRUE)

# Make a histogrem of the total number of steps taken each day
ggplot(data, aes(date, steps)) + geom_bar(stat = "identity",
                                          colour = "steelblue",
                                          fill = "steelblue") +
        facet_grid(.~month, scales = "free") +
        labs(title = "Histogram of Total Number of Steps Taken Each Day", x = "Date",
y = "Total Number of Steps")
```

```
## Warning: Removed 2304 rows containing missing values (position_stack).
```



```
# What is mean and median of total number of steps taken per day
mean(StepsByday)
```

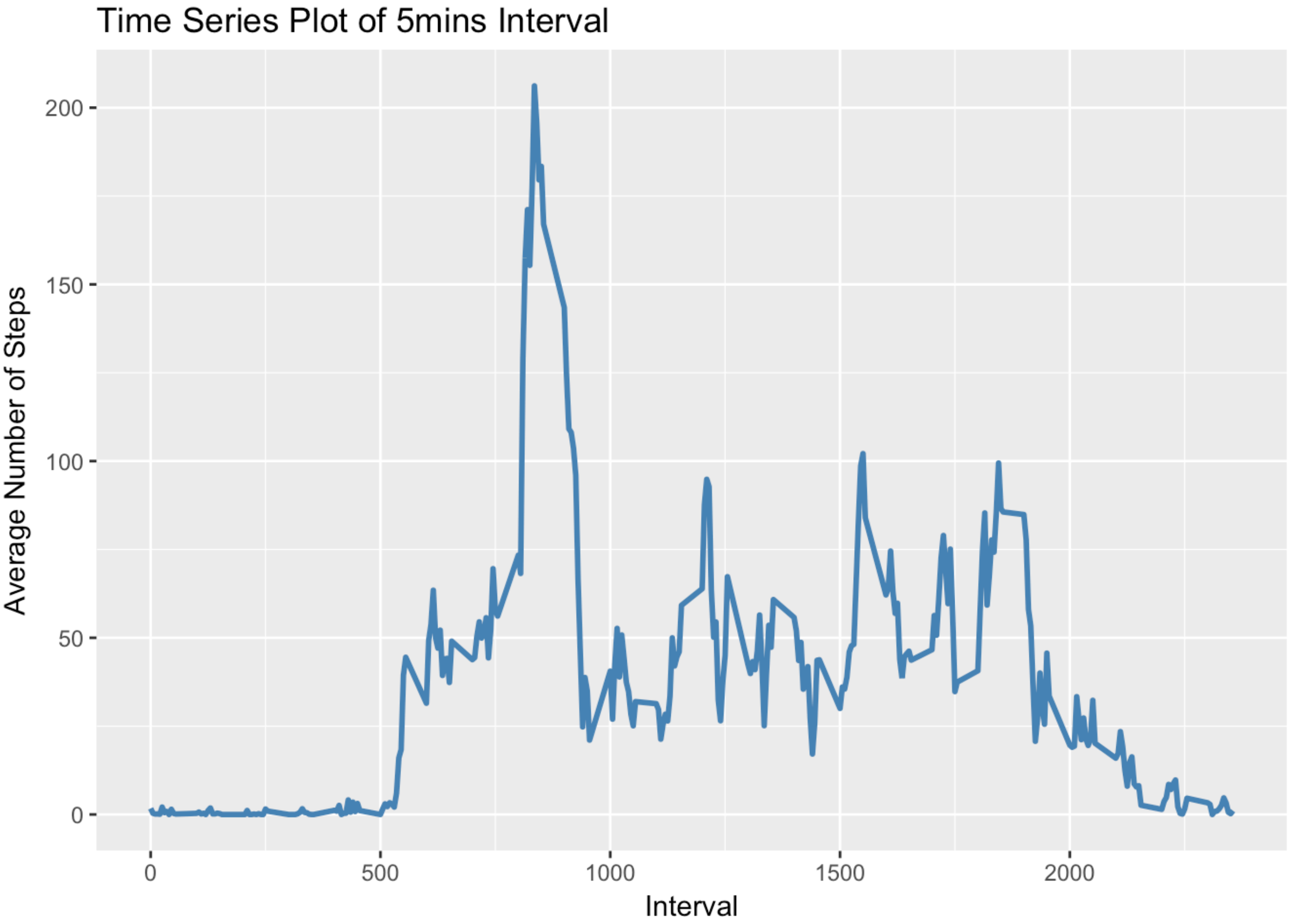
```
## [1] 9354.23
```

```
median(StepsByday)
```

```
## [1] 10395
```

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)
AvgSteps <- aggregate(data$steps,
                      list(interval = as.numeric(as.character(data$interval))),
                      FUN = "mean",
                      na.rm = TRUE)
names(AvgSteps)[2] <- "AvgSteps"
ggplot(AvgSteps, aes(interval, AvgSteps)) +
        geom_line(color = "steelblue", size = 1.0) +
        labs(title = "Time Series Plot of 5mins Interval", x = "Interval", y = "Avera
ge Number of Steps")
```



```
# Which 5-minute interval, on average across all the days in the dataset, contains th
e maximum number of steps?
AvgSteps[AvgSteps$AvgSteps == max(AvgSteps$AvgSteps),]
```

```
##     interval AvgSteps
## 104      835 206.1698
```

Imputing the missing values

```
# check total number of rows with NAs
sum(is.na(data))
```

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

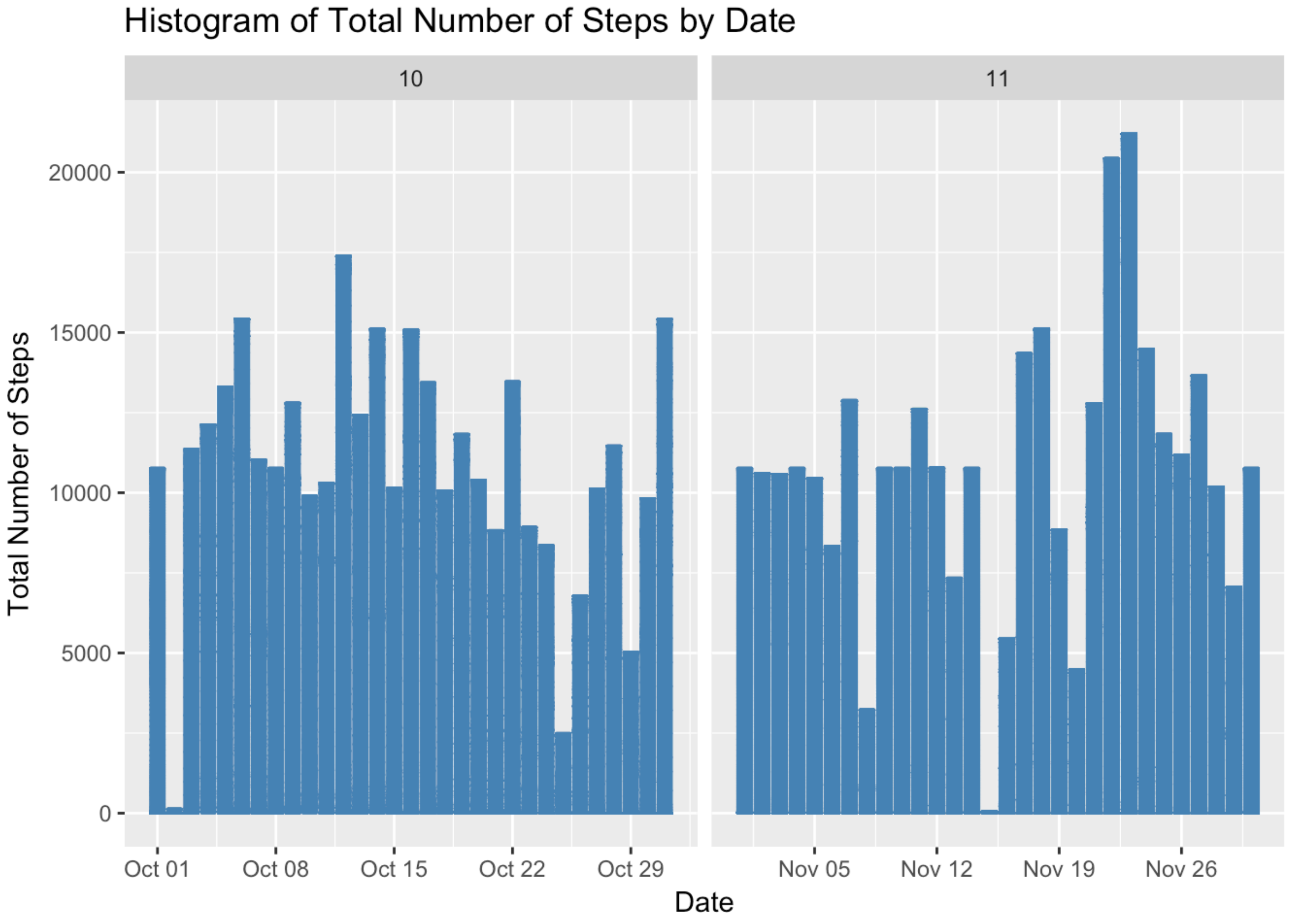
```
# Create a new dataset that is equal to the original dataset but with the missing dat
a filled in.
# fill in missing data by the mean for that 5 mins interval
newdata <- data
for (i in 1:nrow(newdata)) {
        if (is.na(newdata$steps[i])) {
                newdata$steps[i] <- AvgSteps[which(newdata$interval[i] == AvgSteps$in
terval), ]$AvgSteps
        }
}
head(newdata)
```

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

```
sum(is.na(newdata))
```

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

```
# 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.
ggplot(newdata, aes(date, steps)) +
        geom_bar(stat = "identity",
                 color = "steelblue",
                 fill = "steelblue",
                 width = 0.8) +
        facet_grid(.~month, scales = "free") +
        labs(title = "Histogram of Total Number of Steps by Date", x = "Date", y = "Total Number of Steps")
```



```
# mean and median of steps taken of new data with missing values filled
NewTotalSteps <- tapply(newdata$steps, newdata$date, sum, na.rm = TRUE)
mean(NewTotalSteps)
```

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

```
median(NewTotalSteps)
```

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

```
# compare with old mean and median
mean(NewTotalSteps) - mean(StepsByday)
```

```
## [1] 1411.959
```

```
median(NewTotalSteps) - median(StepsByday)
```

```
## [1] 371.1887
```

Are there differences in activity patterns between weekdays and weekends?

```
# Create a new factor variable in the dataset with two levels -- "weekday" and "weekend" indicating whether a given date is a weekday or weekend day.
newdata$daytype <- factor(format(newdata$date, "%A"))
levels(newdata$daytype) <- list(weekdays = c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"),
                                weekend = c("Saturday", "Sunday"))
levels(newdata$daytype)
```

```
## [1] "weekdays" "weekend"
```

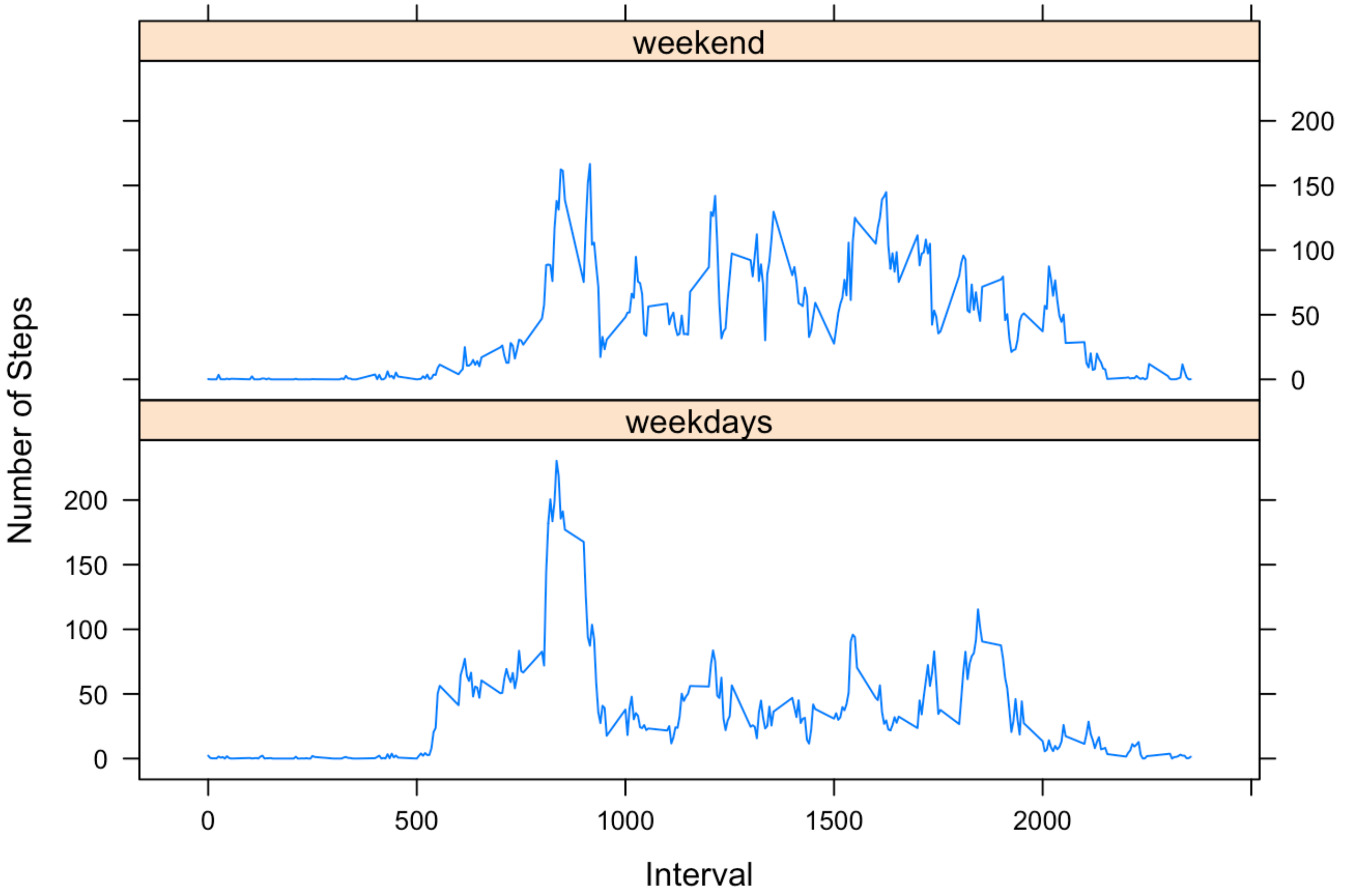
```
table(newdata$daytype)
```

```
## 
## weekdays  weekend 
##    12960     4608
```

```
# Make 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).
AvgStepsDaytype <- aggregate(newdata$steps,
                             list(interval = as.numeric(as.character(newdata$interval)),
                                  daytype = newdata$daytype),
                                FUN = "mean")
names(AvgStepsDaytype)[3] <- "MeanSteps"
head(AvgStepsDaytype)
```

```
##   interval  daytype  MeanSteps
## 1        0 weekdays 2.25115304
## 2        5 weekdays 0.44528302
## 3       10 weekdays 0.17316562
## 4       15 weekdays 0.19790356
## 5       20 weekdays 0.09895178
## 6       25 weekdays 1.59035639
```

```
library(lattice)
xyplot(AvgStepsDaytype$MeanSteps ~ AvgStepsDaytype$interval | AvgStepsDaytype$daytype,
       layout = c(1,2), type = "l",
       xlab = "Interval", ylab = "Number of Steps")
```


weekend
weekdays
Number of Steps
Interval
0
50
100
150
200
0
500
1000
1500
2000