# PA1_template - updated

*Jim Carson*

*March 26, 2017*

# John Hopkins Data Science Specialization

# Course 5: Reproducible Research

# Course Project 1

This assignment uses data from a personal activity monitoring device Device collects data at 5 minute intervals through out the day Two months of data from an anonymous individual collected Oct-Nov 2012

Dataset variables (17,568 observations) - steps: Number of steps taking in a 5-minute interval (missing values are coded as NA) - date: The date on which the measurement was taken in YYYY-MM-DD format - interval: Identifier for the 5-minute interval in which measurement was taken

```
# Set working directory
setwd("~/R/John Hopkins Data Science Specialization/Course 5 - Reproducible Research/Data")

# Load graphics library
library("ggplot2", lib.loc="~/R/win-library/3.3")
library("lattice", lib.loc="~/R/win-library/3.3")

# Read data and review
Step_Data <- read.csv("activity.csv", header = TRUE)
str(Step_Data)
```

```
## '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 ...
```

```
head(Step_Data)
```

```
##   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
```

```
class(Step_Data$date)
```

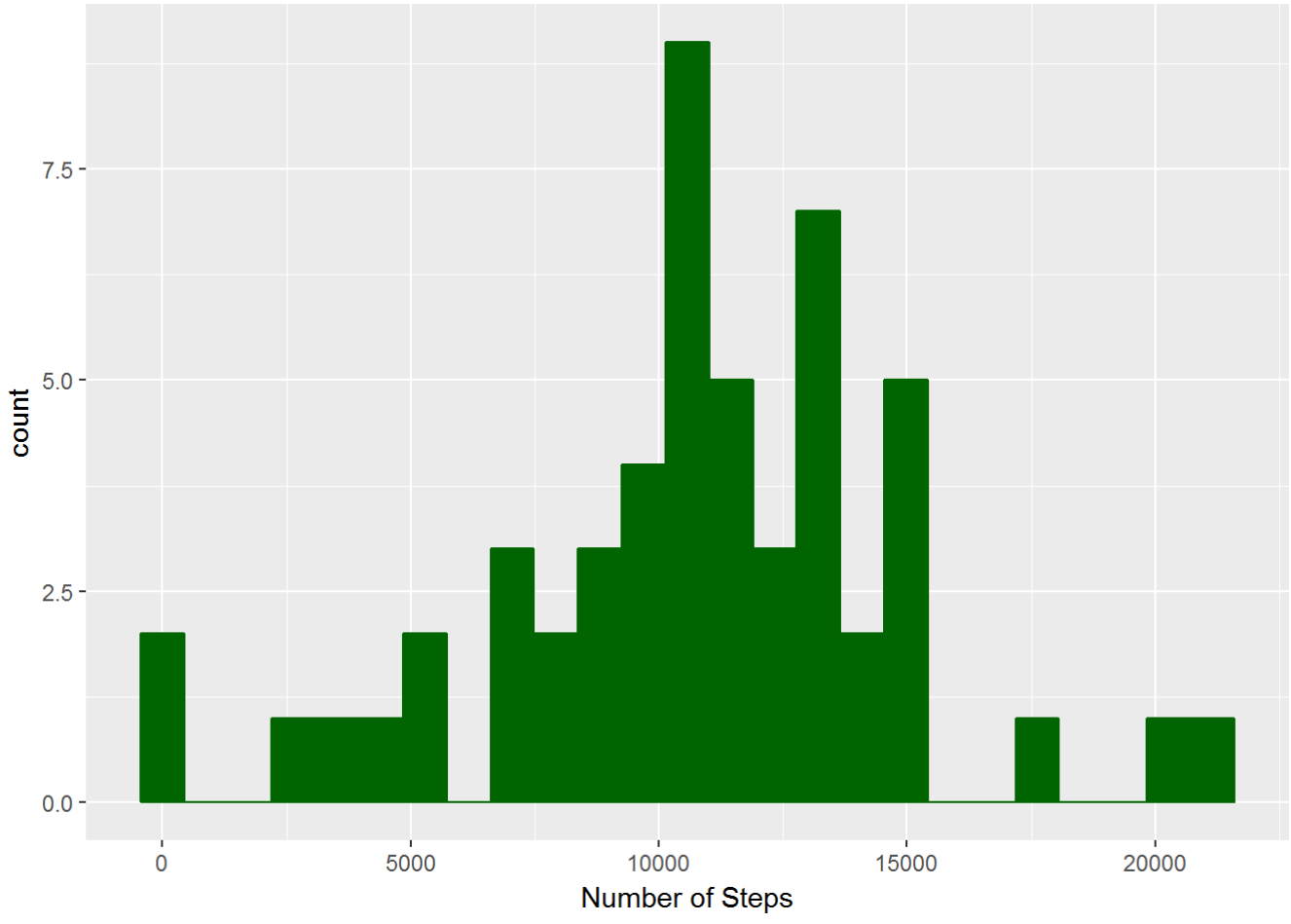
```
## [1] "factor"
```

```
# Since 'date' field from factor to date
Step_Data$date <- as.Date(Step_Data$date)
```

# What is the mean total number of steps taken per day (ignore missing values)?

Calculate the total number of steps taken per day Make a histogram of the total number of steps taken each day Calculate and report the mean and median of the total number of steps taken per day

```
Steps_per_day <- aggregate(steps ~ date, Step_Data, sum)
qplot(Steps_per_day$steps, geom="histogram", bins = 25,  fill=I("dark green"), col=I("dark gree
n"), xlab="Number of Steps")
```




```
Mean_Steps_per_day <- round(mean(Steps_per_day$steps), 0)
Median_Steps_per_day <- round(median(Steps_per_day$steps), 0)
Mean_Steps_per_day
```

```
## [1] 10766
```

```
Median_Steps_per_day
```

```
## [1] 10765
```

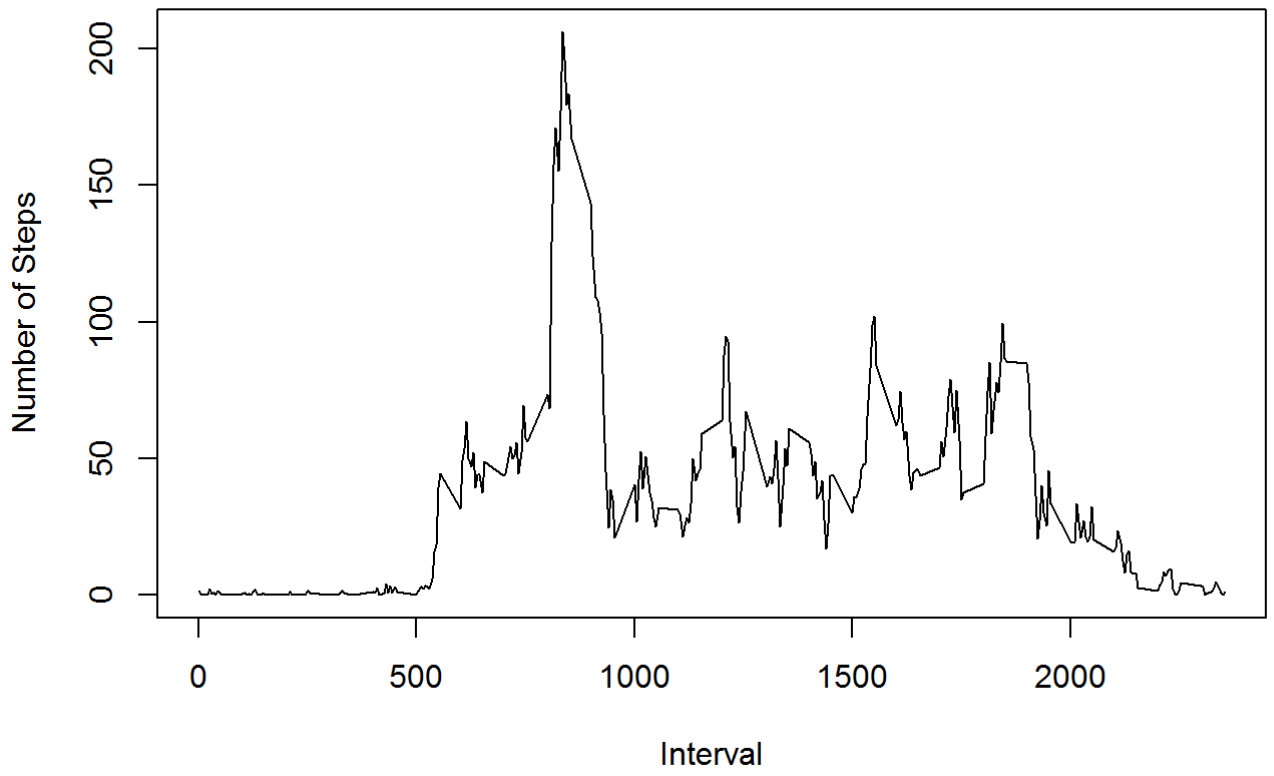
The mean steps per day is 1.076610^{4} and the median steps per day is 1.076510^{4}.

# 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)

# Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

```
Steps_Interval <- aggregate(steps ~ interval, Step_Data, mean)
plot(Steps_Interval$interval,Steps_Interval$steps, type="l", xlab="Interval", ylab="Number of St
eps")
```




```
Interval_Max <- Steps_Interval[which.max(Steps_Interval$steps),1]
Interval_Min <- Steps_Interval[which.min(Steps_Interval$steps),1]
Interval_Max
```

```
## [1] 835
```

```
Interval_Min
```

```
## [1] 40
```

The 5-minute interval with the maximum steps is 835.

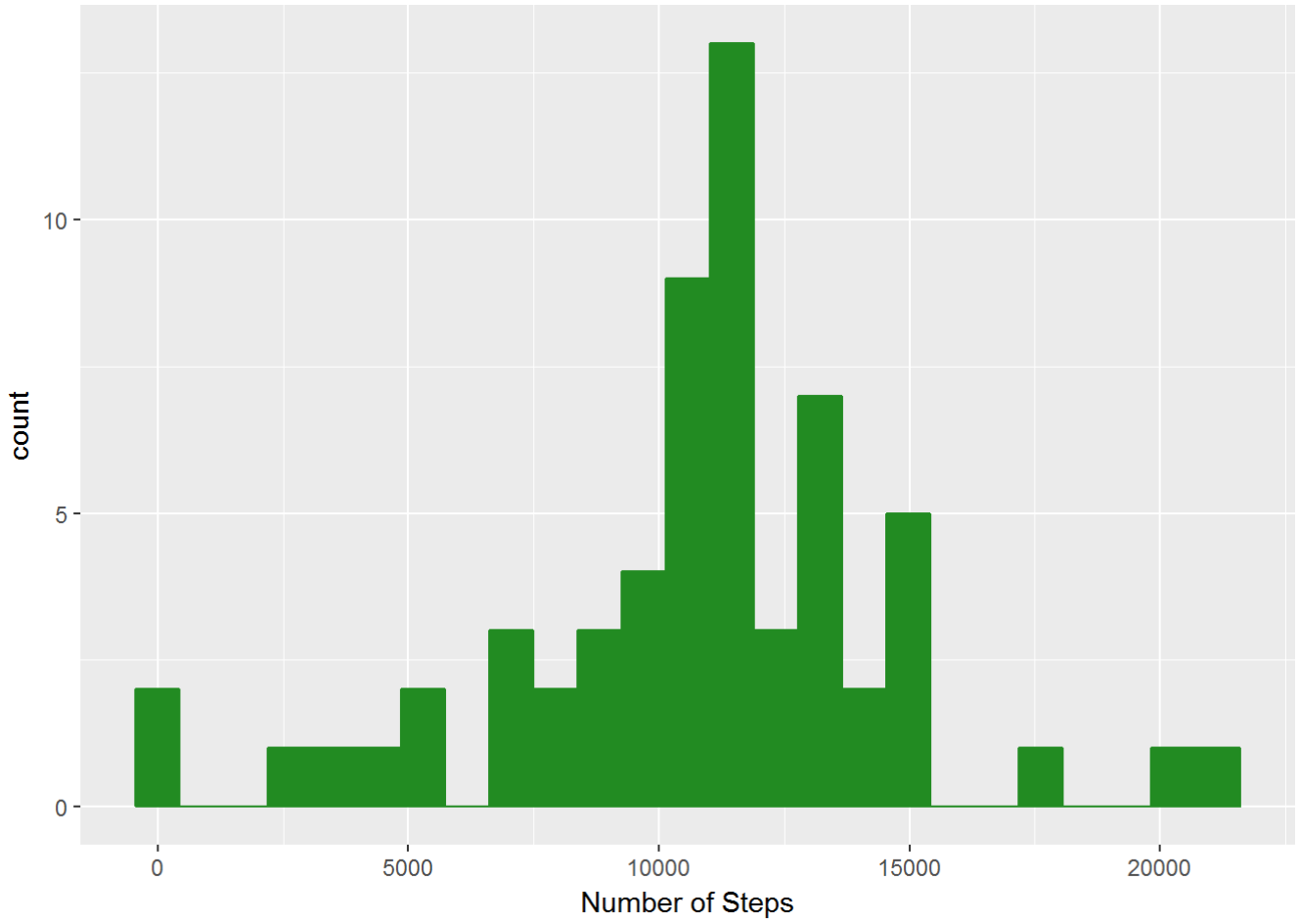
# Impute missing values

Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs) Devise a strategy for filling in all of the missing values in the dataset. Create a new dataset that is equal to the original dataset but with the missing data filled in. 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?

```
Missing_Count <- sum(!complete.cases(Step_Data))
Step_Data_Imputed <- transform(Step_Data, steps = ifelse(is.na(Step_Data$steps), Interval_Min, S
tep_Data$steps))
str(Step_Data_Imputed)
```

```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  40 40 40 40 40 40 40 40 40 40 ...
##  $ date    : Date, format: "2012-10-01" "2012-10-01" ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

```
Steps_per_day_Imp <- aggregate(steps ~ date, Step_Data_Imputed, sum)
qplot(Steps_per_day_Imp$steps, geom="histogram", bins = 25,  fill=I("forest green"), col=I("fore
st green"), xlab="Number of Steps")
```




```
Mean_Steps_per_day_Imputed <- mean(Steps_per_day_Imp$steps)
Median_Steps_per_day_Imputed <- median(Steps_per_day_Imp$steps)

Mean_Diff <- round(Mean_Steps_per_day_Imputed - Mean_Steps_per_day, 0)
Median_Diff <- round(Median_Steps_per_day_Imputed - Median_Steps_per_day, 0)
Total_Diff <- round(sum(Steps_per_day_Imp$steps) - sum(Steps_per_day$steps), 0)
Mean_Diff
```

```
## [1] 99
```

```
Median_Diff
```

```
## [1] 693
```

```
Total_Diff
```

```
## [1] 92160
```

The difference in mean steps is 99, the difference in median steps is 693, and the total difference is 9.21610^{4}.

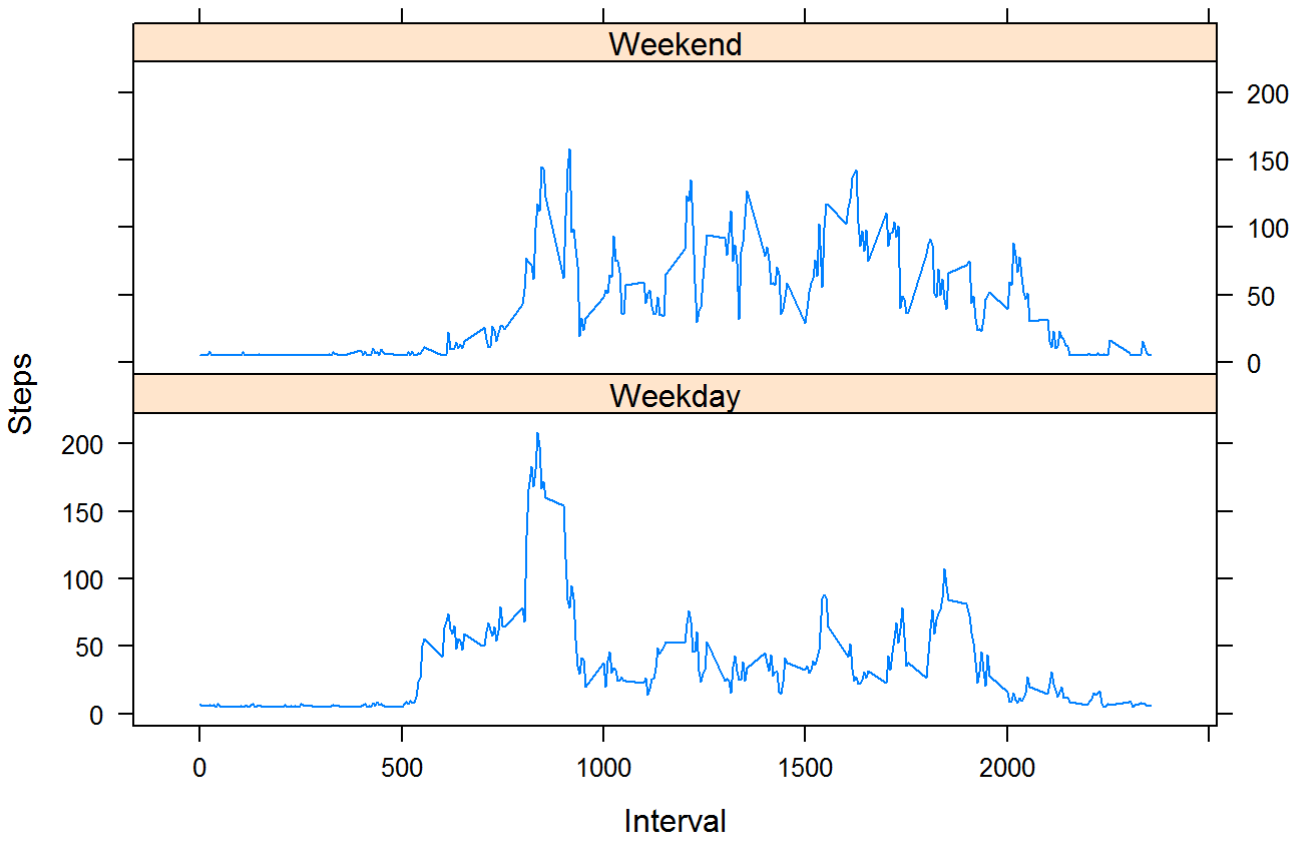
# Are there differences in activity patterns between weekdays and weekends?

The weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part. 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. 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). See the README file in the GitHub repository to see an example of what this plot should look like using simulated data.

```
weekdays <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")
Step_Data_Imputed$Weekday =
as.factor(ifelse(is.element(weekdays(as.Date(Step_Data_Imputed$date)),weekdays), "Weekday", "Wee
kend"))

Steps_Interval_Imputed <- aggregate(steps ~ interval + Weekday, Step_Data_Imputed, mean)

xyplot(Steps_Interval_Imputed$steps ~ Steps_Interval_Imputed$interval|Steps_Interval_Imputed$Wee
kday, main="Average Steps per Day by Interval",xlab="Interval", ylab="Steps",layout=c(1,2),
type="l")
```


Average Steps per Day by Interval
Weekend
Weekday
Steps
Interval
0
50
100
150
200
0
500
1000
1500
2000