R Markdown file for Reproducible Research Week 2

First steps is to load the data and look at some basic summary statistics of the data

```
setwd("//LanFspd01.ncsbn.org/UserProfileFolders$/LErickson/Documents/Coursera/data")
Activity <- read.csv("activity.csv", header = TRUE, na.strings="NA")
summary(Activity)
```

```
##      steps                date          interval     
##  Min.   :  0.00   10/1/2012 :  288   Min.   :   0.0  
##  1st Qu.:  0.00   10/10/2012:  288   1st Qu.: 588.8  
##  Median :  0.00   10/11/2012:  288   Median :1177.5  
##  Mean   : 37.38   10/12/2012:  288   Mean   :1177.5  
##  3rd Qu.: 12.00   10/13/2012:  288   3rd Qu.:1766.2  
##  Max.   :806.00   10/14/2012:  288   Max.   :2355.0  
##  NA's   :2304     (Other)   :15840
```

```
rdate <- as.Date(Activity$date,"%m/%d/%y")
head(Activity)
```

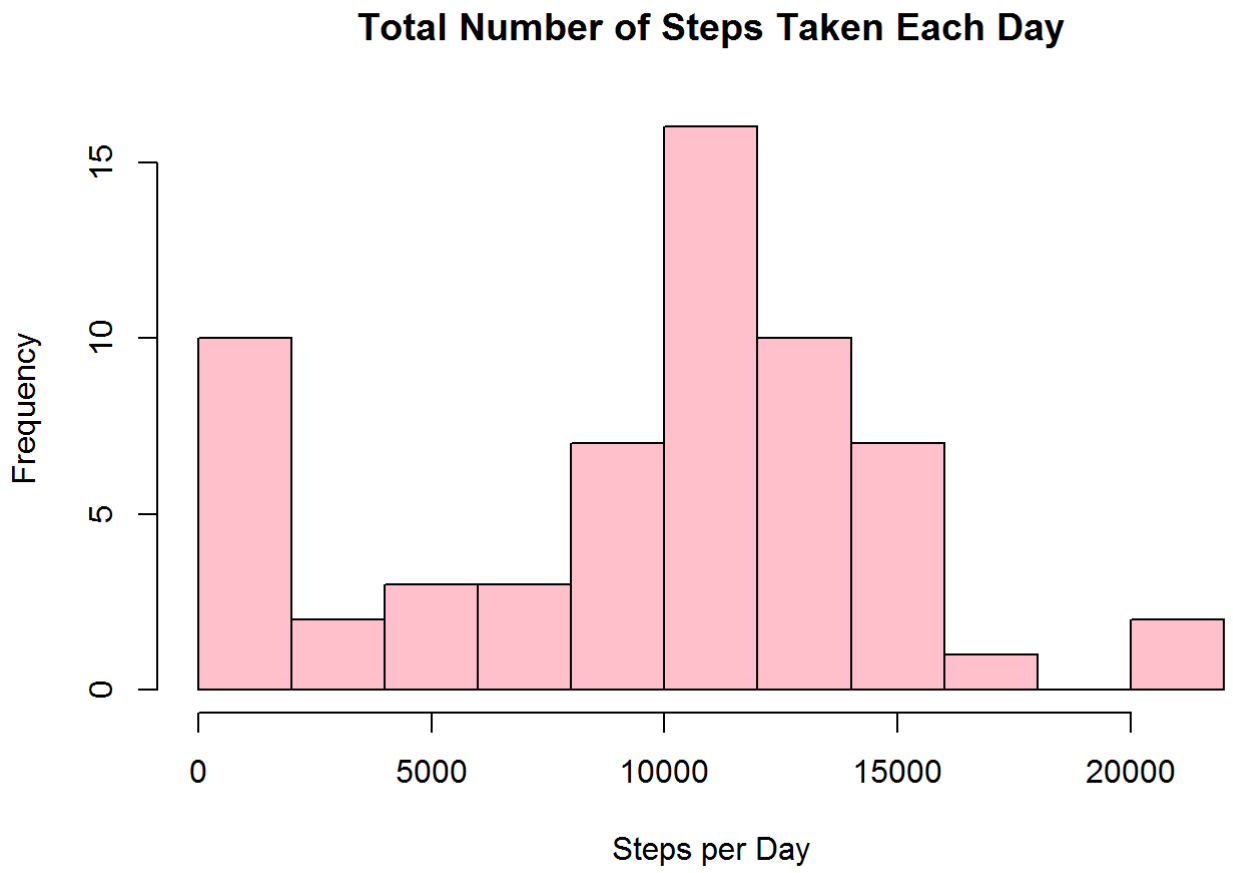
```
##   steps      date interval
## 1    NA 10/1/2012        0
## 2    NA 10/1/2012        5
## 3    NA 10/1/2012       10
## 4    NA 10/1/2012       15
## 5    NA 10/1/2012       20
## 6    NA 10/1/2012       25
```

Now, I want to look at a **histogram** of the total number of steps taken each day.

To do this, I need to:

```
    1. Calculate the sum of the steps taken each day
    2. Create the histogram of the calculated steps
```

```
TotalDailySteps <- tapply(Activity$steps, Activity$date, FUN = sum, na.rm = TRUE)
hist(TotalDailySteps, breaks=10, xlab="Steps per Day", col="pink", main="Total Number of Steps T
aken Each Day")
```



Next, I want to to know what the **mean** and **median** number of steps taken each day and rounded to one digit

```
options(digits=1)  ##round to one digit
stepsmean <- mean(TotalDailySteps, na.rm = TRUE)
stepsmedian <- median(TotalDailySteps, na.rm = TRUE)
```

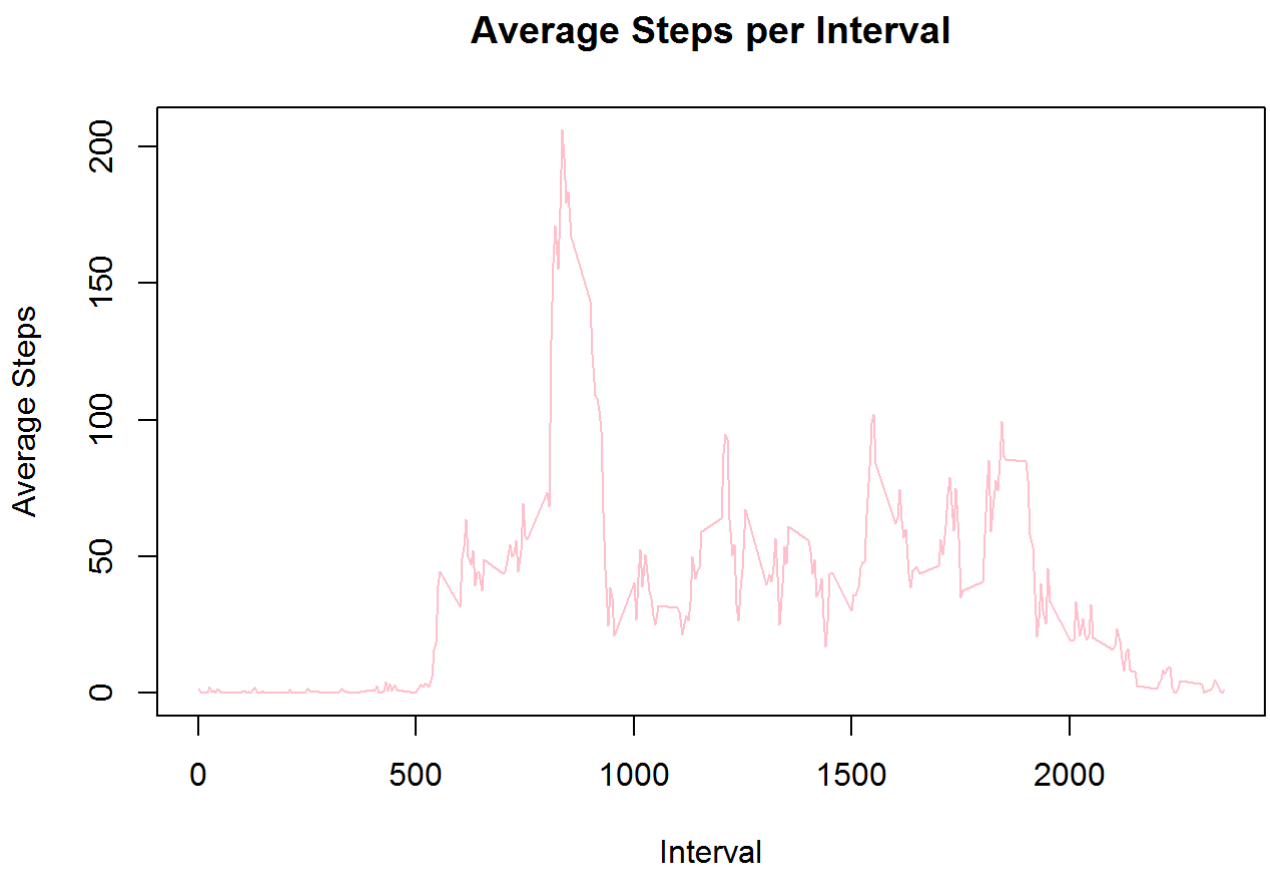
The mean number of steps taken each day is 9354.2.
The median number of steps taken each day is 10395.

I want to view a time series plot of the average number of steps taken per interval.

To do this, I need to:

```
1. Calculate the mean number of steps per interval
2. Create a data frame with the caluclated mean
3. Create the plot of the new data frame as a time series plot
```

```
IntervalStep <- tapply(Activity$steps, Activity$interval, mean, na.rm = TRUE) ##Calculating the
 mean number of steps taken at each interval
IntAvg <- data.frame(interval=as.integer(names(IntervalStep)), avg=IntervalStep) ##creating a da
ta frame from IntervalStep
plot(IntAvg$avg ~ IntAvg$interval, type='l', col="pink", xlab = "Interval", ylab = "Average Step
s", main="Average Steps per Interval")
```



Next, I want to find the maximum number of steps taken during a 5-minute interval

```
StepsMax <- IntAvg[which.max(IntAvg$avg), ]
```

The maximum number of steps taken during a 5-minute interval is 206.2 steps at interval 835

This code shows how I dealt with missing data

first step is to calculate the number of missing data values

```
MissingSteps <- sum(is.na(Activity$steps))  ##to find how many missing data points there are
```

There are 2304 missing data values

Next, I need to apply the median for a 5-minute interval and use that number of fill the missing data values. Here are my steps to do this:

1. Calculate the median number of steps for each interval.
2. Create a data frame from the median calculated.
3. Merge the created data frame with the original data set.
4. Replace the missing values with the calculated median.

```
MedStep <- tapply(Activity$steps, Activity$interval, median, na.rm = TRUE) ##Calculating the mea
n number of steps taken at each interval
IntMed <- data.frame(interval=as.integer(names(MedStep)), step=MedStep) ##creating a data frame
 from MedStep

TempActivity <- merge(Activity, IntMed, by="interval", all.y = FALSE) ##Merge the created data f
rame with the original data set
TempActivity$steps[is.na(TempActivity$steps)] <- as.integer(round(TempActivity$step[is.na(TempAc
tivity$steps)])) ##uses the calculated median in place of the missing values
keeps <- names(Activity)
TempActivity <- TempActivity[keeps]
MissingSteps2 <- sum(is.na(TempActivity$steps))
```

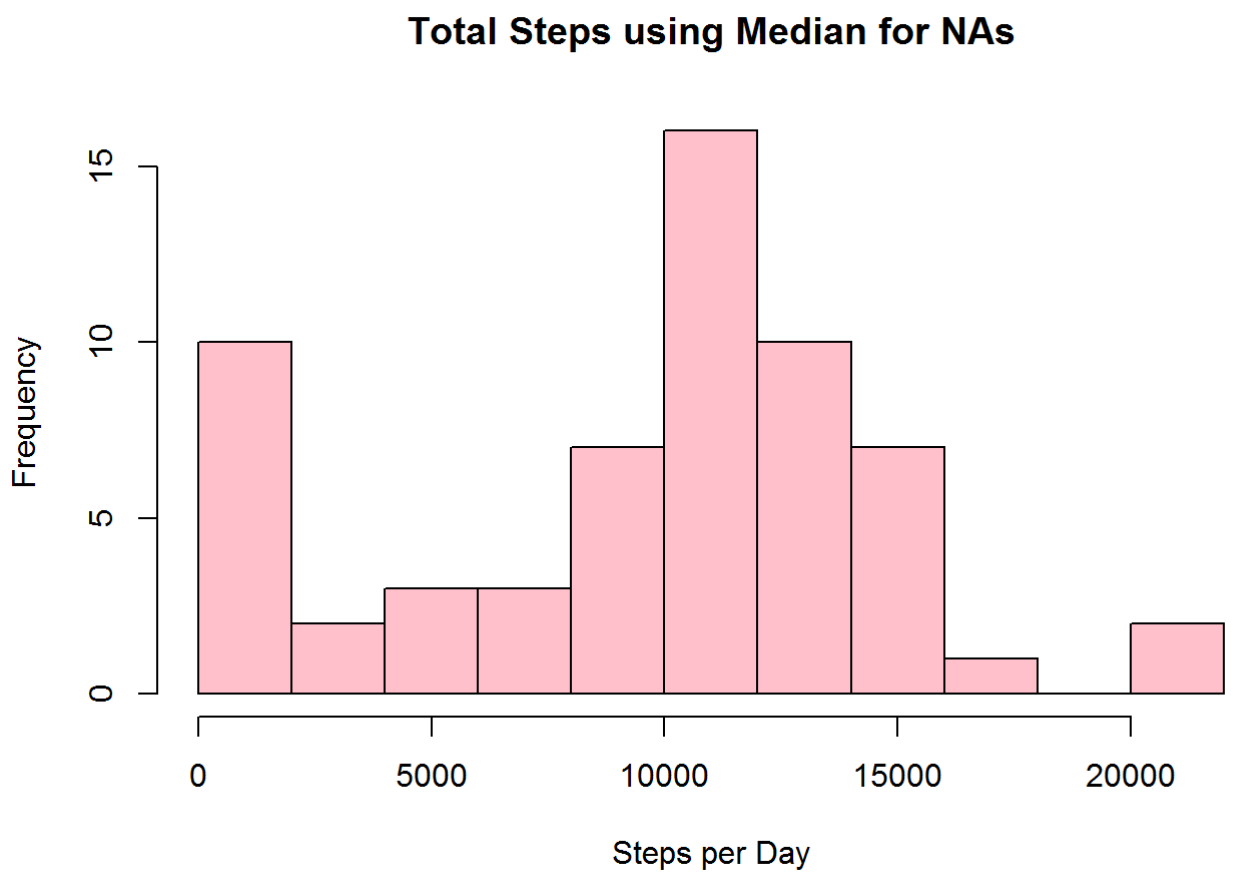
We can see that this method has producted 0 missing data values

Let’s look at a histogram of the total steps taken where the missing data is replaced with the median of the 5-minute interval

To do this, I need to:

1. Calculate the total steps per each day
2. create a histogram of this calculation

```
RevTotalDailySteps <- tapply(TempActivity$steps, TempActivity$date, FUN = sum, na.rm = TRUE)
hist(RevTotalDailySteps, breaks=10, main="Total Steps using Median for NAs", xlab="Steps per Da
y", col="pink")
```



I want to look at the differences in average number of steps taken per 5-minute interval across weekdays and weekends

```
  1. Create a variable that labels each oberservation as a weekend or weekday based on the given
 date
  2. Calculate the mean number of steps per each interval
  3. Load lattice because we want to stack two graphs
  4. Plote the data
```

```
Weekend <- c("Saturday", "Sunday") #Identifying the weekend
TempActivity$weekday = as.factor(ifelse(is.element(weekdays(as.Date(TempActivity$date)),
Weekend), "Weekend",  "Weekday"))  #flesh out the weekends by using an if else statement - if it
 is not a weekend, it is a weekday


IntervalDow <- aggregate(steps ~ interval + weekday, TempActivity, mean)
head(IntervalDow)
```

```
##   interval weekday steps
## 1        0 Weekday  1.72
## 2        5 Weekday  0.34
## 3       10 Weekday  0.13
## 4       15 Weekday  0.00
## 5       20 Weekday  0.08
## 6       25 Weekday  2.09
```

```
library(lattice)
xyplot(IntervalDow$steps ~ IntervalDow$interval|IntervalDow$weekday, layout=c(1,2), type="l", co
l="pink", main="Steps per Interval Comparing Weekday to Weekend", ylab="Steps", xlab="Interval")
```