Reproducible Research - Peer Assessment 1

Loading and Perprocessing the Data

Show any code that is needed to: 1. Load the data (i.e. read.csv()) 2. Process/transform the data (if necessary) into a format suitable for your analysis

```
#First load the libraries needed for these studies
library(dplyr)
```

```
##
## 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(knitr)
library(ggplot2)
#Load the data file into a data frame
activity <- read.csv("activity.csv")
#Clean the dataset by removing NA values
cleanData <- activity[ with(activity,{!(is.na(steps))}),]
```

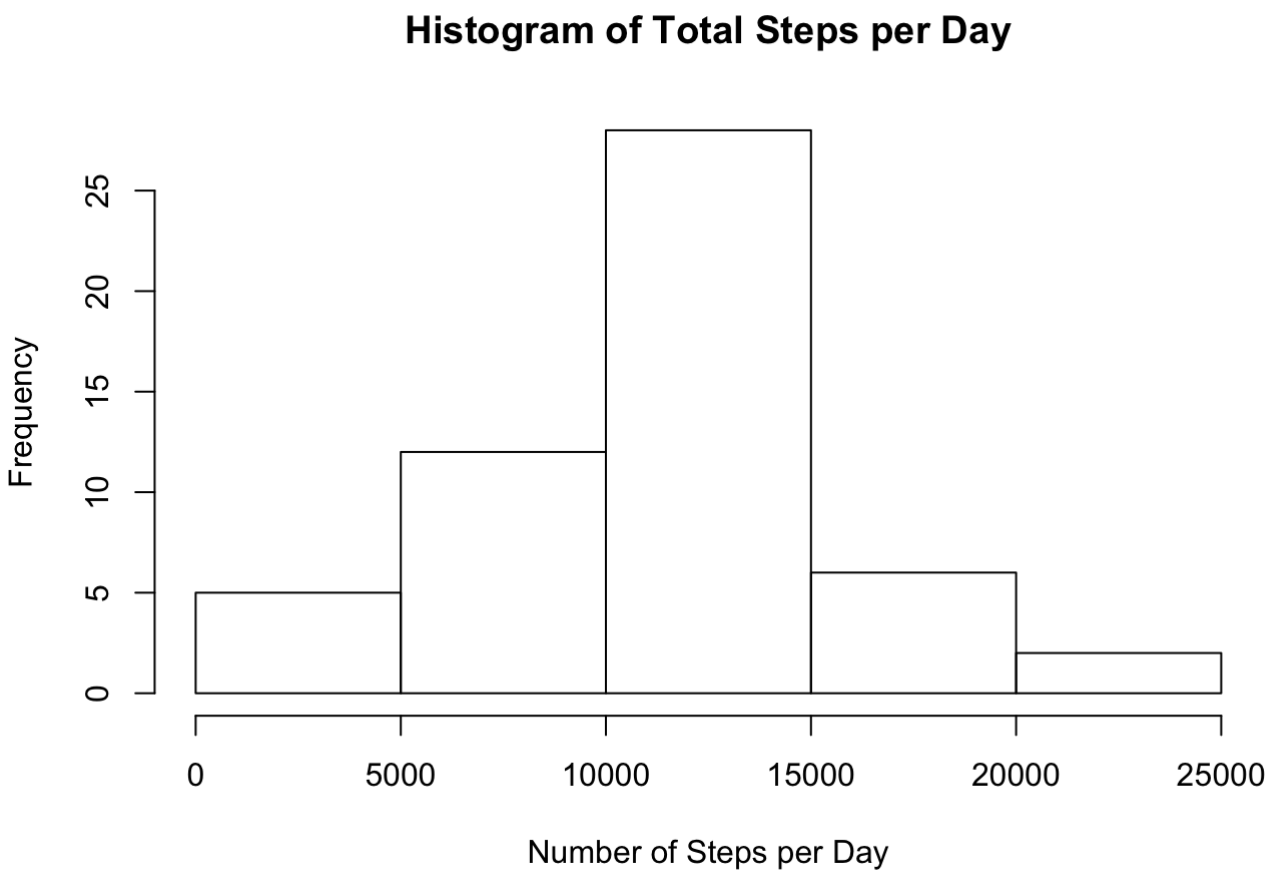
What is Mean Total Number of Steps Taken per Day?

For this part of the assignment, you can ignore the missing values in the dataset.

1. Calculate the total number of steps taken per day
2. If you do not understand the difference between a histogram and a barplot, research the difference between them. Make a histogram of the total number of steps taken each day
3. Calculate and report the mean and median of the total number of steps taken per day

```
#Calculate the total number of steps taken per day
totalSteps <-aggregate(steps ~ date, cleanData, sum)
```

```
#Create a Histogram of the total steps taken per day
hist(totalSteps$steps, main = "Histogram of Total Steps per Day", xlab = "Number of Step
s per Day")
```




```
#Calculate and report the mean and median of total steps per day
summary(totalSteps)
```

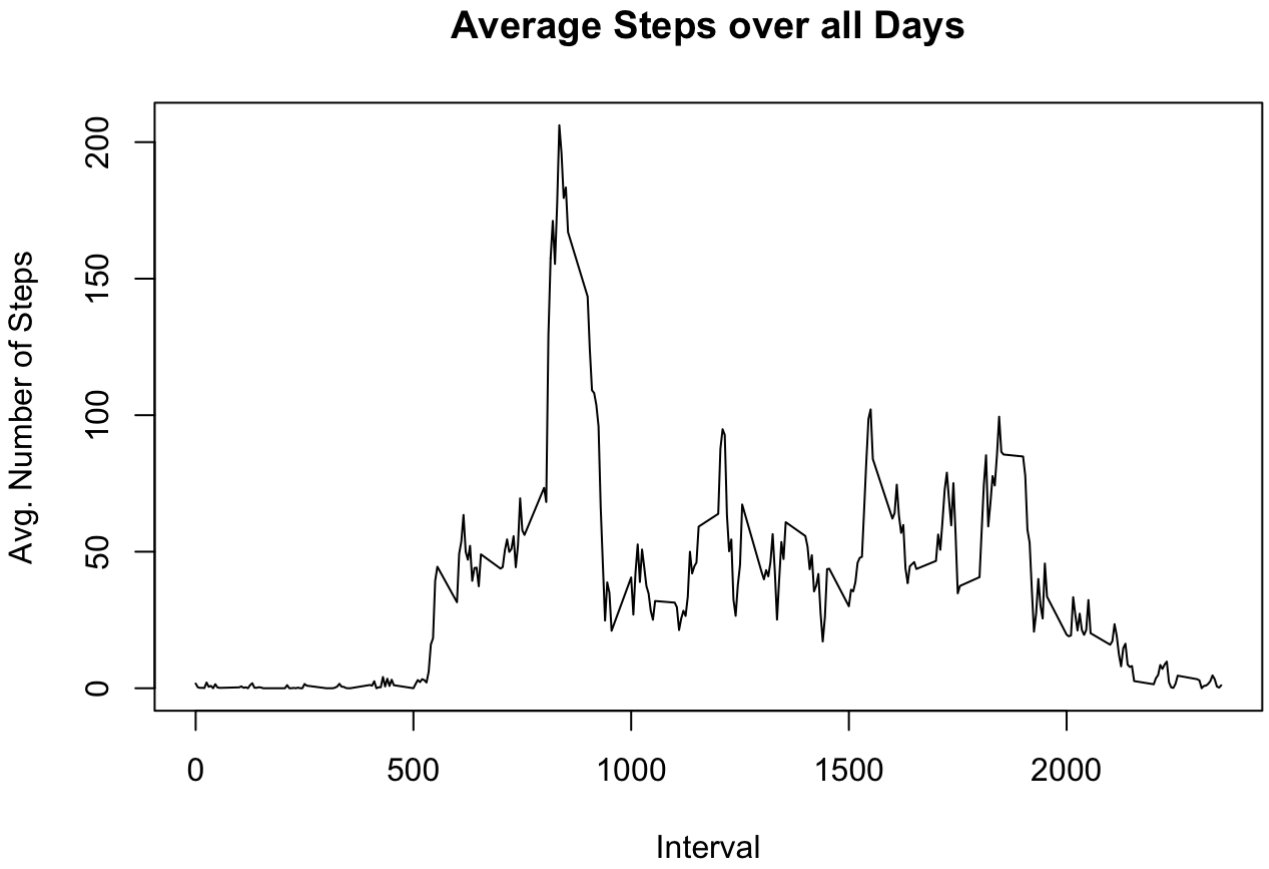
```
##          date        steps
##  2012-10-02: 1   Min.   :   41
##  2012-10-03: 1   1st Qu.: 8841
##  2012-10-04: 1   Median :10765
##  2012-10-05: 1   Mean   :10766
##  2012-10-06: 1   3rd Qu.:13294
##  2012-10-07: 1   Max.   :21194
##  (Other)   :47
```

The mean number of steps taken per day is 10,766, and the median is 10,765

What is the average daily activity pattern?

1. 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)
2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

```
#Plot time series of the average number of steps taken across all days
meanInterval <- aggregate(steps ~ interval, data = cleanData, FUN="mean")
plot(meanInterval$interval,meanInterval$steps, type ="l", main = "Average Steps over all
Days", xlab ="Interval", ylab = "Avg. Number of Steps")
```




```
#Find the max number of steps
maxSteps <- which.max(meanInterval$steps)
meanInterval[maxSteps,]
```

```
##     interval    steps
## 104      835 206.1698
```

Imputing Missing Values

Note that there are a number of days/intervals where there are missing values (coded as NA). The presence of missing days may introduce bias into some calculations or summaries of the data.

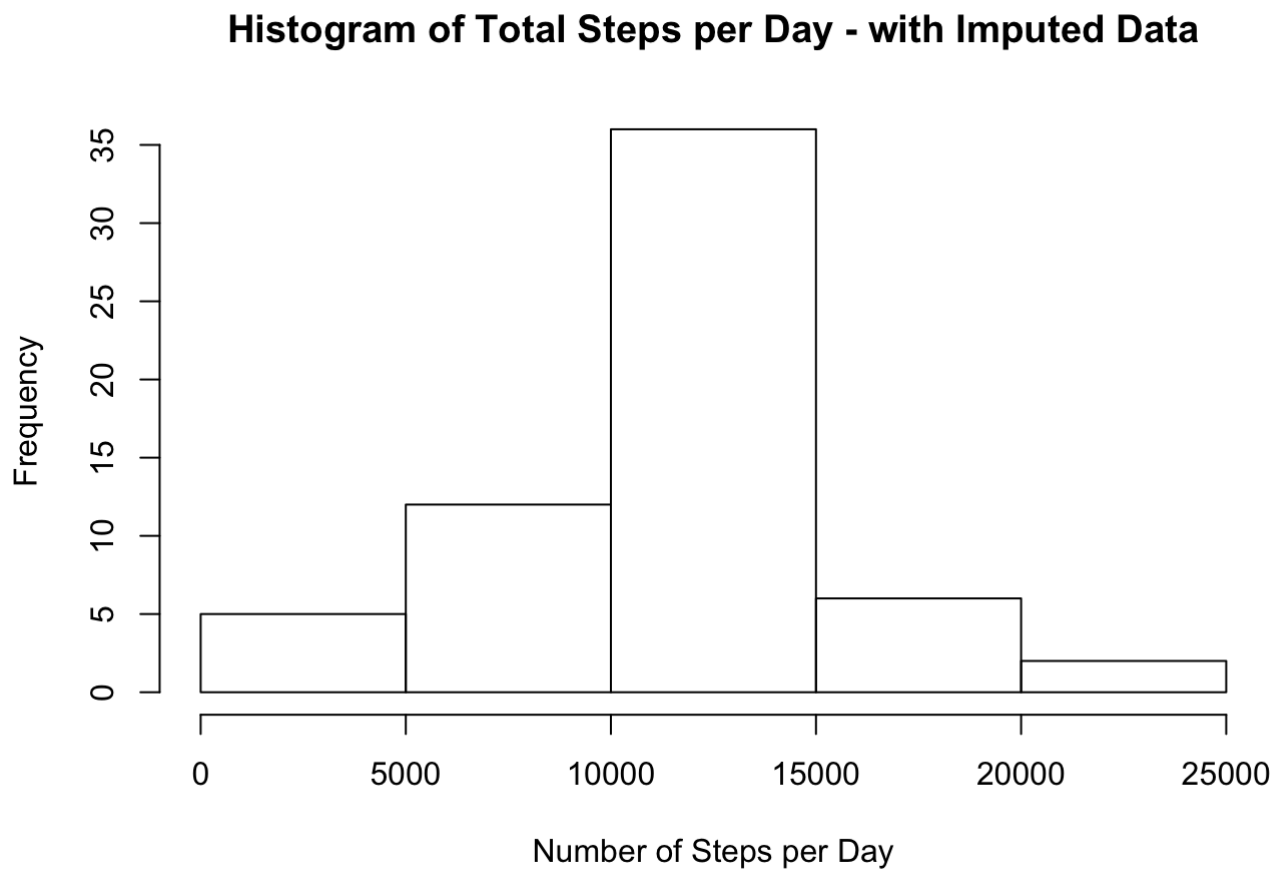
1. Calculate and report the total number of missing values in the dataset (i.e. the total number of rows with NAs)
2. Devise a strategy for filling in all of the missing values in the dataset. The strategy does not need to be sophisticated. For example, you could use the mean/median for that day, or the mean for that 5-minute interval, etc.
3. Create a new dataset that is equal to the original dataset but with the missing data filled in.

4. 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?

```
#Calculate and report missing values
sum(is.na(activity))
```

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

```
#Impute data to original dataframe
imputedActivity <- activity
for (i in 1:nrow(imputedActivity)) {
  if (is.na(imputedActivity$steps[i])) {
    intSteps <- imputedActivity$interval[i]
    newSteps <- meanInterval[
      meanInterval$interval == intSteps,]
    imputedActivity$steps[i] <- newSteps$steps
  }
}
#Create new dataset with missing data filled in and make a histogram
imputedActivity_Day <- aggregate(steps ~ date, imputedActivity, sum)
hist(imputedActivity_Day$steps, main = "Histogram of Total Steps per Day - with Imputed
 Data", xlab = "Number of Steps per Day")
```



Imputing value strategy: The NA values of the dataset were replaced by imputing values using the mean of the 5-minute intervals.

```
#Calculate the mean and median and compare vs. the original data
summary(imputedActivity_Day)
```

```
##          date        steps
##  2012-10-01: 1   Min.   :   41
##  2012-10-02: 1   1st Qu.: 9819
##  2012-10-03: 1   Median :10766
##  2012-10-04: 1   Mean   :10766
##  2012-10-05: 1   3rd Qu.:12811
##  2012-10-06: 1   Max.   :21194
##  (Other)   :55
```

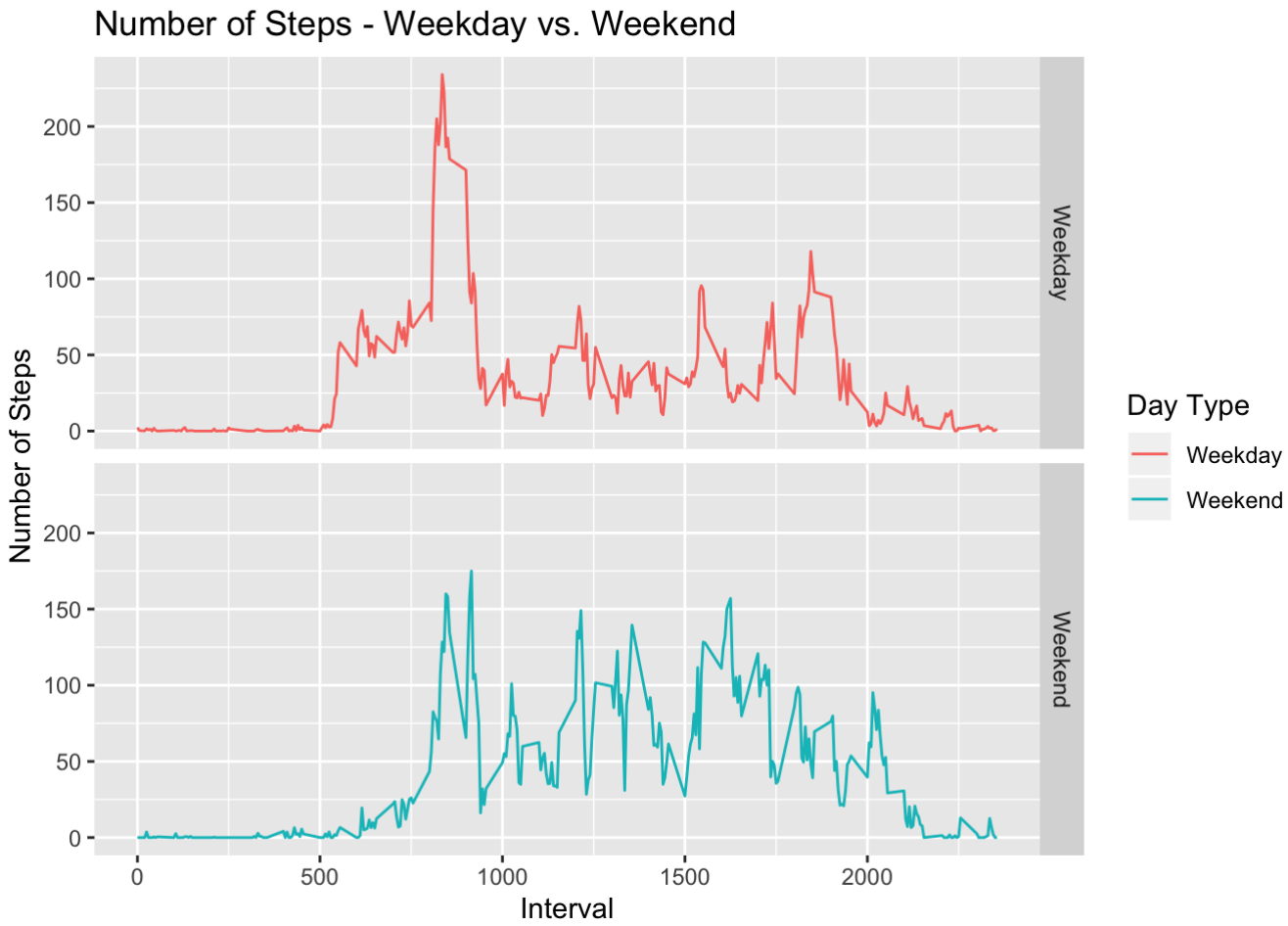
Although the mean and median for the imputed activity dataset are only different by ~1-step vs. the original dataset, the dispersion is wider as the 1st and 3rd quartiles are larger by ~500-1000 steps.

Are there Differences in Activity Patterns between Weekdays and Weekends?

For this part the weekdays() function may be of some help here. Use the dataset with the filled-in missing values for this part.

1. 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.
2. 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.

```
#Create a factor variable in the dataset for the two day types
weekDay <- function(day) {
    days <- weekdays(as.Date(day, "%Y-%m-%d"))
    if(!(days == "Saturday" || days == "Sunday")) {
        week <- "Weekday"
    } else {
        week <- "Weekend"
    }
    week
}
activity$dayWeek <- as.factor(sapply(activity$date, weekDay))
stepsDay <- aggregate(steps ~ interval+dayWeek, activity, FUN = "mean")
#Create a panel plot
ggplot(stepsDay, aes(interval, steps)) + geom_line(stat = "identity", aes(color = dayWee
k))+facet_grid(dayWeek~., scales = "fixed", space = "fixed") + labs(x="Interval", y="Num
ber of Steps") + ggtitle("Number of Steps - Weekday vs. Weekend") + labs(color='Day Typ
e')
```



The test subject is more active earlier in the day on weekdays vs. weekend days, but less active through the middle of the day. My guess is they have a normal ‘desk’ job during the week while they spend their weekends catching up on housework and chores.