# Reproducible Research Project 1

```
#load required packages
library(plyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

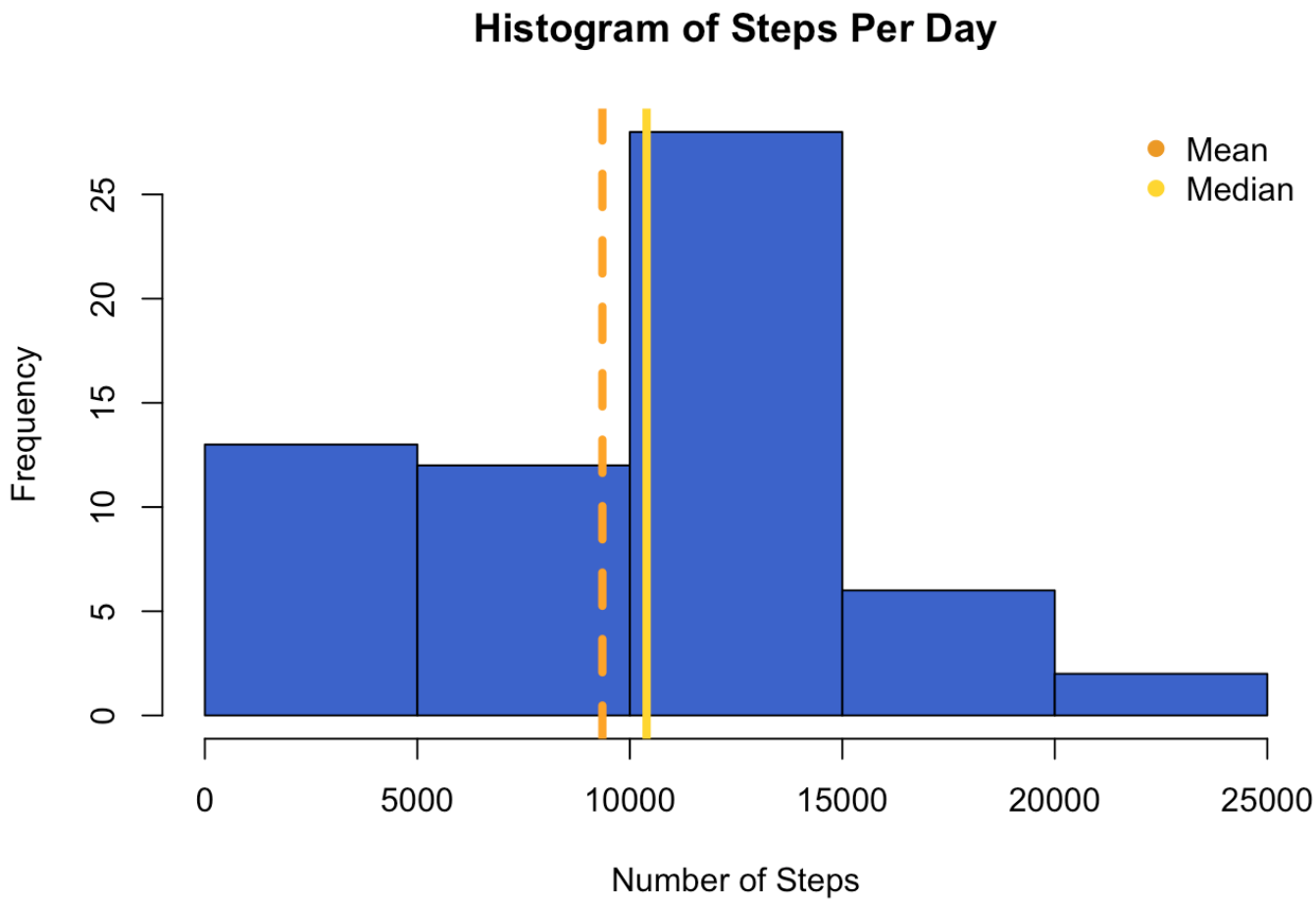
```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#load the data and process
setwd("~/Downloads")
activity<-read.csv("activity.csv")
activity$date<-as.Date(activity$date)
```

```
steps.per.day<-ddply(activity,.(date),summarize,steps=sum(steps,na.rm=T))
par(mfrow=c(1,1))
hist(steps.per.day$steps,xlab="Number of Steps",ylab="Frequency",main="Histogram o
f Steps Per Day",col="royalblue3")
abline(v=mean(steps.per.day$steps),lwd=4,col="orange",lty=2)
abline(v=median(steps.per.day$steps),lwd=4,col="gold")
legend("topright",c("Mean","Median"),col=c("orange2","gold"),bty="n",pch=19)
```

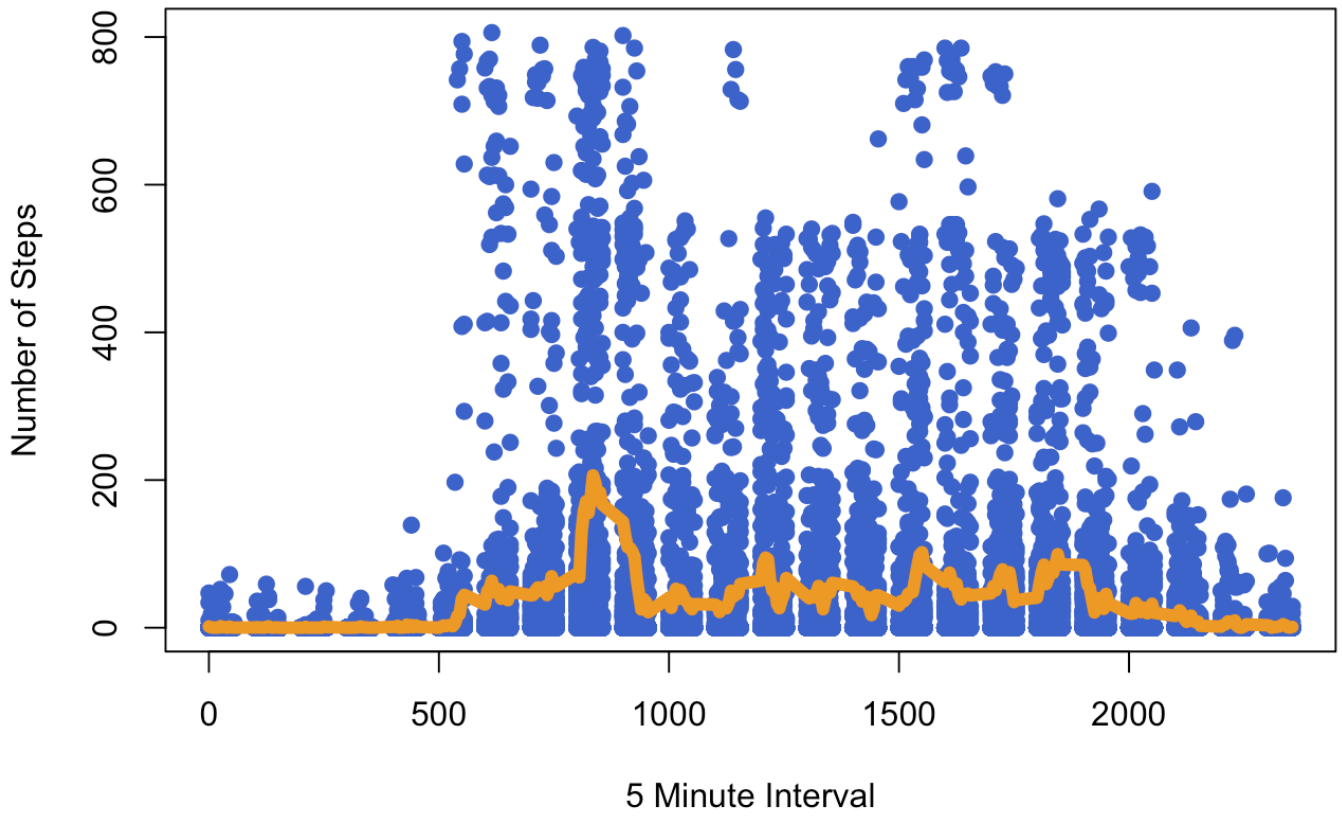



```
avg.steps.per.day<-tapply(activity$steps,activity$interval,mean,na.rm=T)
```

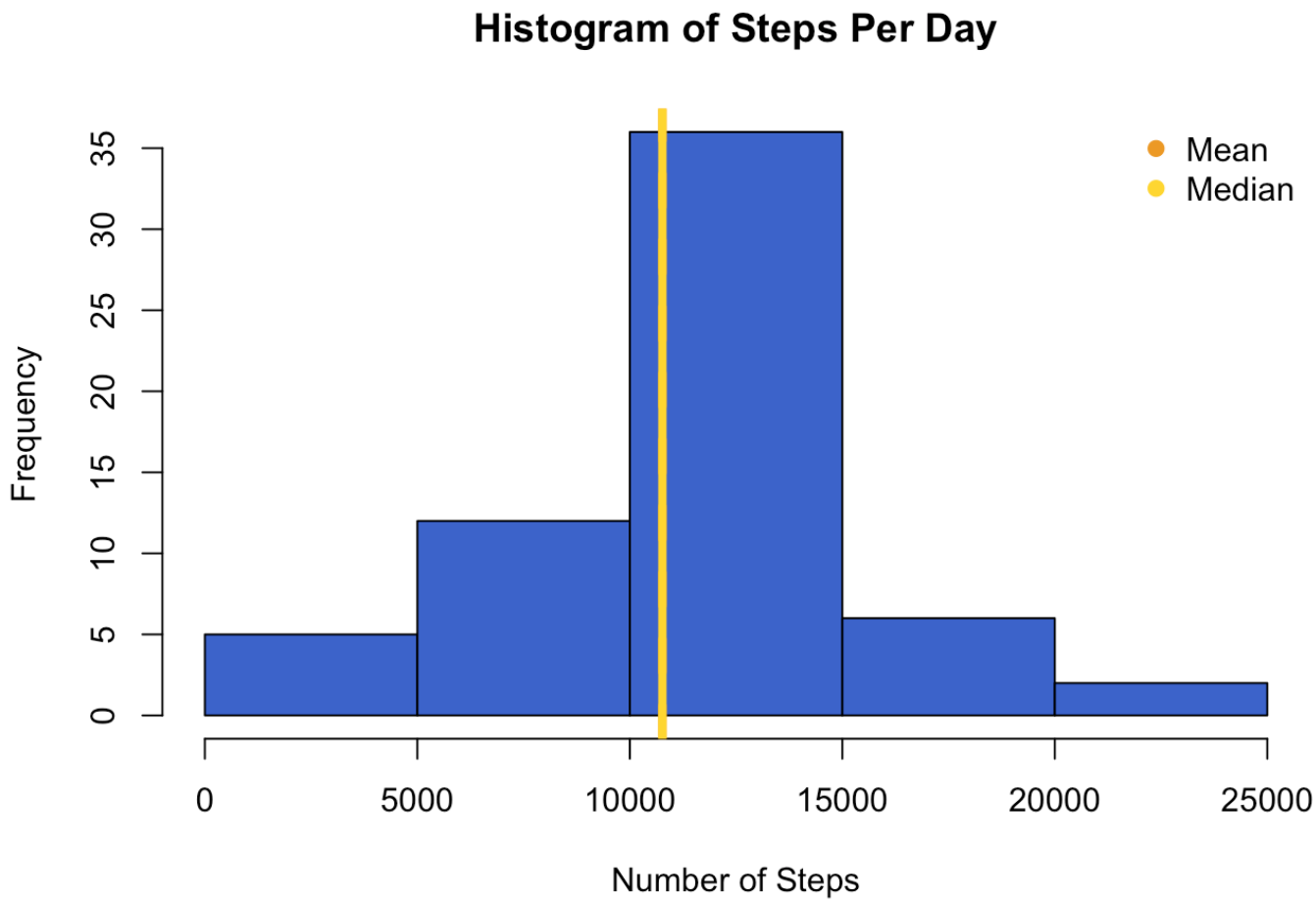
The plot above shows the histogram of total steps across the days in the data set. There are vertical lines for the mean and median. Note missing values are excluded from the data, but may lead to biased results if certain days have more missing values since we are calculating total steps. The mean total number of steps per day is 9354.2295082, and the median is 10395.

Here is the average number of steps taken in each five minute interval (averaged across days), which is depicted in the orange line. The blue points show the actual values, and there is high variance in some intervals. This plot omits missing values, which can skew values if data is missing non randomly across intervals. The five minute interval with the largest average number of steps is 835.

```
par(mfrow=c(1,1))
plot(activity$interval,activity$steps,pch=19,col="royalblue3",xlab="5 Minute Inter
val",ylab="Number of Steps")
lines(names(avg.steps.per.day),avg.steps.per.day,type="l",col="orange2",lwd=6)
```




```
activity.imputed<-activity
mean.interval<-tapply(activity$steps,activity$interval,mean,na.rm=T)
for(i in 1:nrow(activity.imputed)) {
  if(is.na(activity.imputed[i,1])) {
    activity.imputed$steps[i]<-mean.interval[names(mean.interval)==activity.impute
d$interval[i]]
  }
}
steps.per.day2<-ddply(activity.imputed,.(date),summarize,steps=sum(steps,na.rm=T))
hist(steps.per.day2$steps,xlab="Number of Steps",ylab="Frequency",main="Histogram
of Steps Per Day",col="royalblue3")
abline(v=mean(steps.per.day2$steps),lwd=4,col="orange",lty=2)
abline(v=median(steps.per.day2$steps),lwd=4,col="gold")
legend("topright",c("Mean","Median"),col=c("orange2","gold"),bty="n",pch=19)
```




Now we input missing values in the data using the average value of steps taken on that interval. This averages across all days and therefore is only valid if the distribution of steps taken is the same across days. The histogram of total steps taken across all days is recreated below with the imputed data. Filling in the data changes the mean and median such that they are now equal, and both increase to 1.076618910^{4}.

We turn to the final question that uses the inputed data to compare activity patterns across weekdays and weekends. The plot below shows the average number of steps in each five minute interval separately for weekends and weekdays. As the plot shows there is not too much difference in activity patterns for weekends and weekdays.

```
activity.imputed$weekday<-weekdays(activity.imputed$date)
activity.imputed$weekday[activity.imputed$weekday=="Sunday"|activity.imputed$weekd
ay=="Saturday"]<-"weekend"
activity.imputed$weekday[!activity.imputed$weekday%in%"weekend"]<-"weekday"
par(mfrow=c(2,1))
avg.steps.per.weekday<-tapply(activity.imputed$steps[activity.imputed$weekday=="we
ekday"],activity$interval[activity.imputed$weekday=="weekday"],mean,na.rm=T)
avg.steps.per.weekend<-tapply(activity.imputed$steps[activity.imputed$weekday=="we
ekend"],activity$interval[activity.imputed$weekday=="weekend"],mean,na.rm=T)
plot(names(avg.steps.per.weekday),avg.steps.per.weekday,type="l",col="royalblue
2",lwd=6,ylab="Number of Steps",xlab="Weekday")
plot(names(avg.steps.per.weekend),avg.steps.per.weekend,type="l",col="orange2",lw
d=6,ylab="Number of Steps",xlab="Weekend")
```