# project1

*Weihua Shen*

*2019/10/16*

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.2
```

```r
library(plyr)
activity <- read.csv("~/Downloads/activity.csv")
activity$day <- weekdays(as.Date(activity$date))
activity$DateTime<- as.POSIXct(activity$date, format="%Y-%m-%d")
clean <- activity[!is.na(activity$steps),]
sumTable <- aggregate(activity$steps ~ activity$date, FUN=sum, )
colnames(sumTable)<- c("Date", "Steps")
hist(sumTable$Steps, breaks=5, xlab="Steps", main = "Total Steps per Day")
```
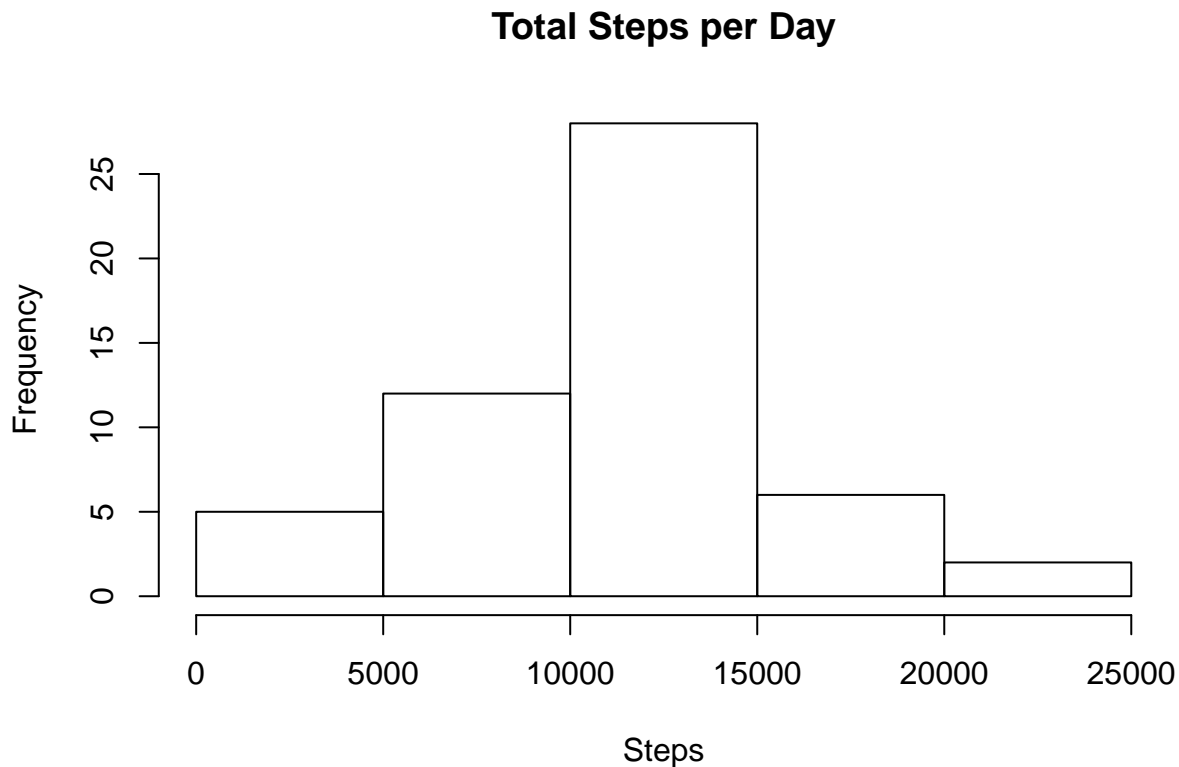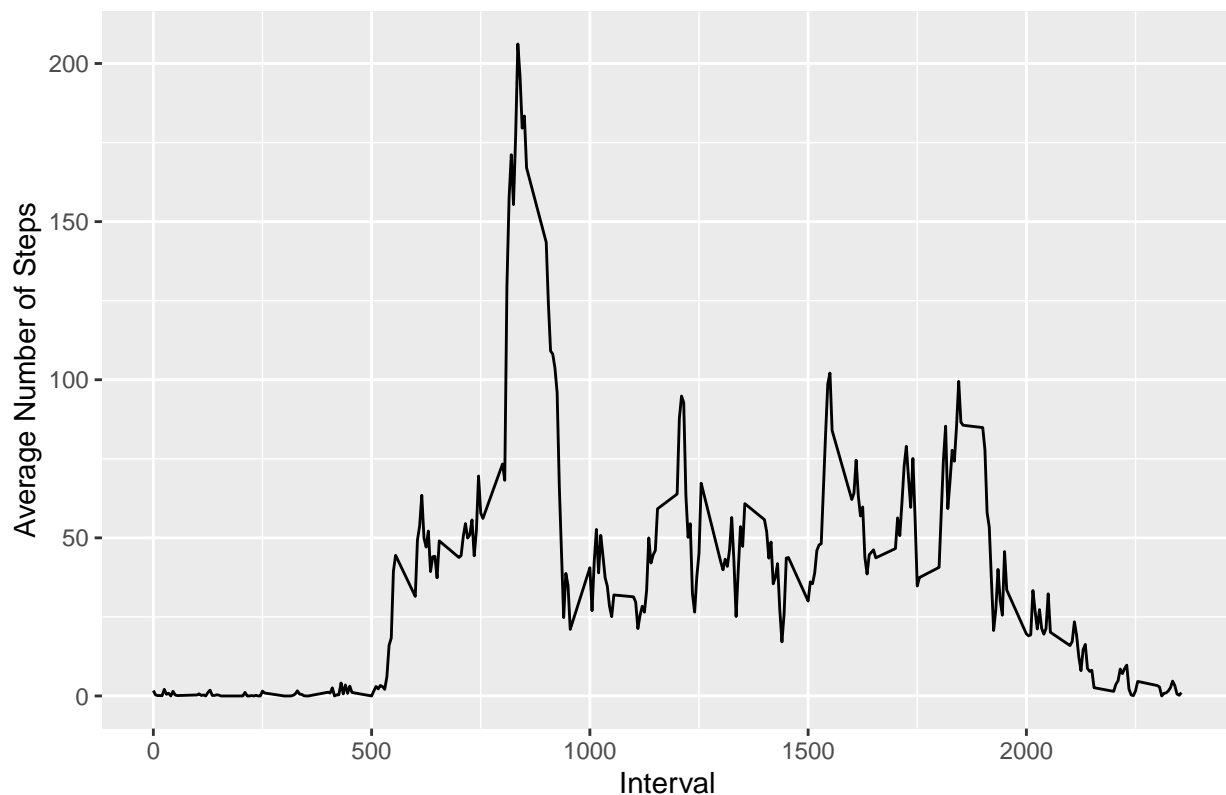


```r
as.integer(mean(sumTable$Steps))
```

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

```r
as.integer(median(sumTable$Steps))
```

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

```r
clean <- activity[!is.na(activity$steps),]
intervalTable <- ddply(clean, .(interval), summarize, Avg = mean(steps))
p <- ggplot(intervalTable, aes(x=interval, y=Avg), xlab = "Interval", ylab="Average Number of Steps")
p + geom_line()+xlab("Interval")+ylab("Average Number of Steps")+ggtitle("Average Number of Steps per I
```

## Average Number of Steps per Interval



```
maxSteps <- max(intervalTable$Avg)
intervalTable[intervalTable$Avg==maxSteps,1]
```

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

```
nrow(activity[is.na(activity$steps),])
```

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

```
avgTable <- ddply(clean, .(interval, day), summarize, Avg = mean(steps))
nadata<- activity[is.na(activity$steps),]
newdata<-merge(nadata, avgTable, by=c("interval", "day"))
newdata2<- newdata[,c(6,4,1,2,5)]
colnames(newdata2)<- c("steps", "date", "interval", "day", "DateTime")
mergeData <- rbind(clean, newdata2)
sumTable2 <- aggregate(mergeData$steps ~ mergeData$date, FUN=sum, )
colnames(sumTable2)<- c("Date", "Steps")
as.integer(mean(sumTable2$Steps))
```
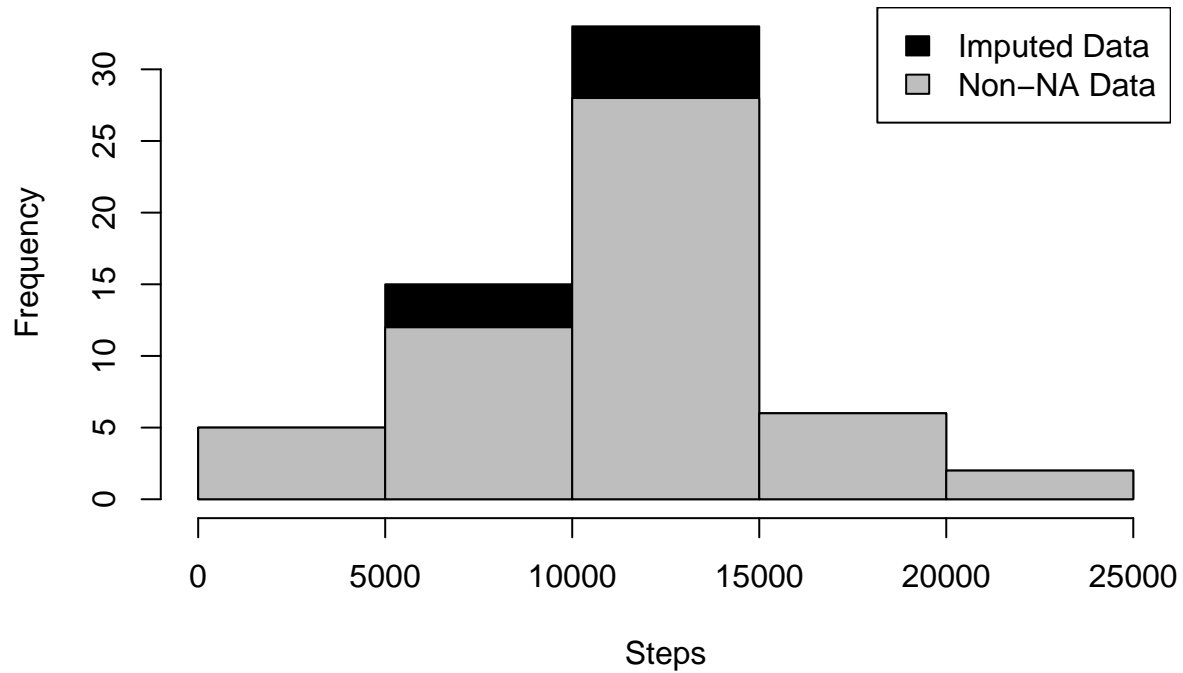
```
## [1] 10821
```

```
as.integer(median(sumTable2$Steps))
```

```
## [1] 11015
```

```
hist(sumTable2$Steps, breaks=5, xlab="Steps", main = "Total Steps per Day with NAs Fixed", col="Black")
hist(sumTable$Steps, breaks=5, xlab="Steps", main = "Total Steps per Day with NAs Fixed", col="Grey", ad
legend("topright", c("Imputed Data", "Non-NA Data"), fill=c("black", "grey") )
```

## Total Steps per Day with NAs Fixed



```
mergeData$DayCategory <- ifelse(mergeData$day %in% c("Saturday", "Sunday"), "Weekend", "Weekday")
library(lattice)
intervalTable2 <- ddply(mergeData, .(interval, DayCategory), summarize, Avg = mean(steps))
xyplot(Avg~interval|DayCategory, data=intervalTable2, type="l",  layout = c(1,2),
       main="Average Steps per Interval Based on Type of Day",
       ylab="Average Number of Steps", xlab="Interval")
```

Average Steps per Interval Based on Type of Day