```
DF <- read.csv(file.choose(),sep=',',header=T,stringsAsFactors = F)
str(DF)
```

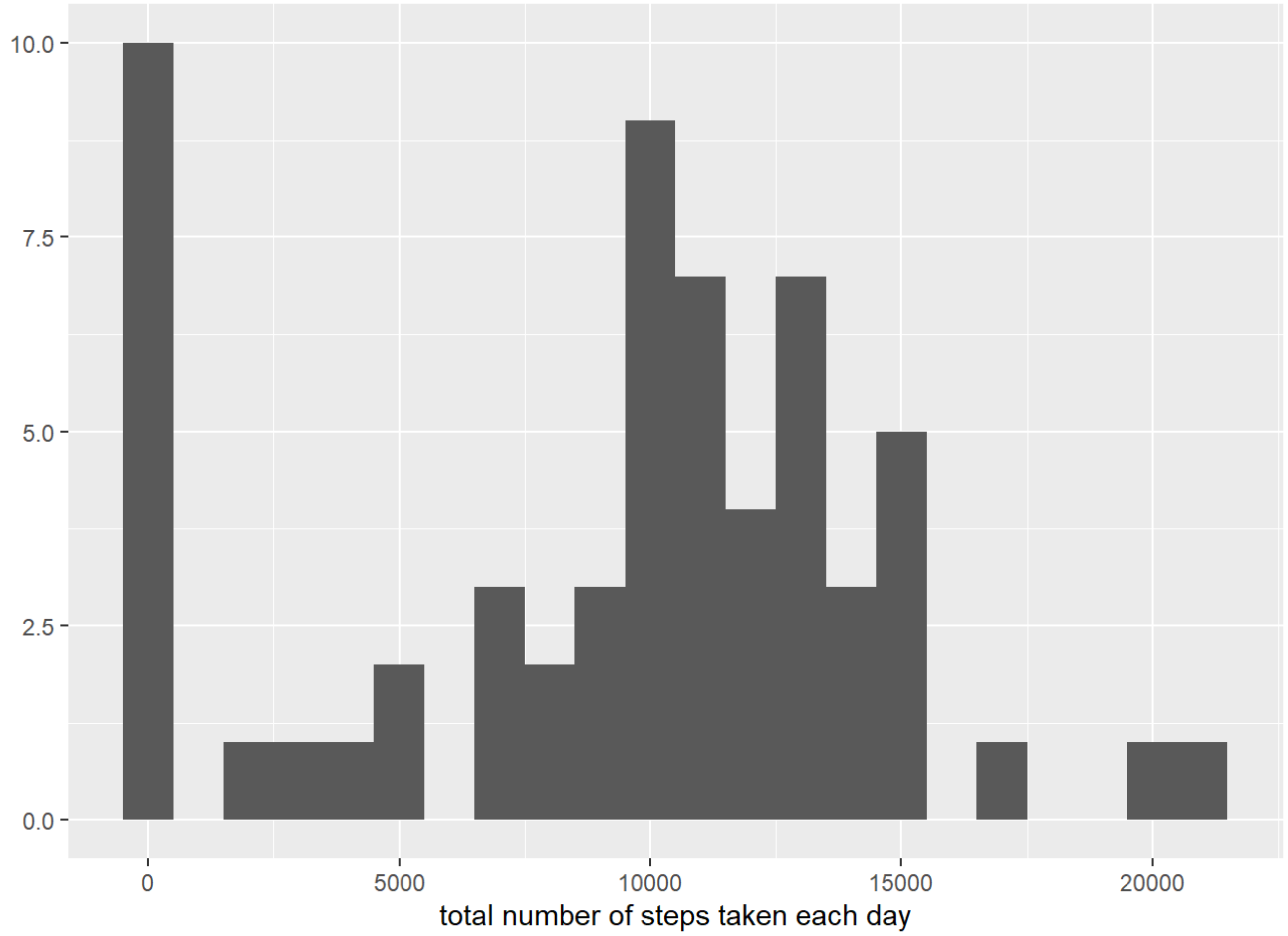
```
## 'data.frame':    17568 obs. of  3 variables:
##  $ steps   : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ date    : chr  "2012-10-01" "2012-10-01" "2012-10-01" "2012-10-01" ...
##  $ interval: int  0 5 10 15 20 25 30 35 40 45 ...
```

```
# Create the sums of steps per date
DFsteps <- tapply(DF$steps, DF$date, FUN=sum, na.rm=TRUE)

# Perform histogram of steps per day
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
qplot(DFsteps, binwidth=1000, xlab="total number of steps taken each day")
```



```
# Create mean and median of steps per day
stepsMean <- mean(DFsteps, na.rm=TRUE)
stepsMedian <- median(DFsteps, na.rm=TRUE)

# Output mean and median
stepsMean
```

```
## [1] 9354.23
```

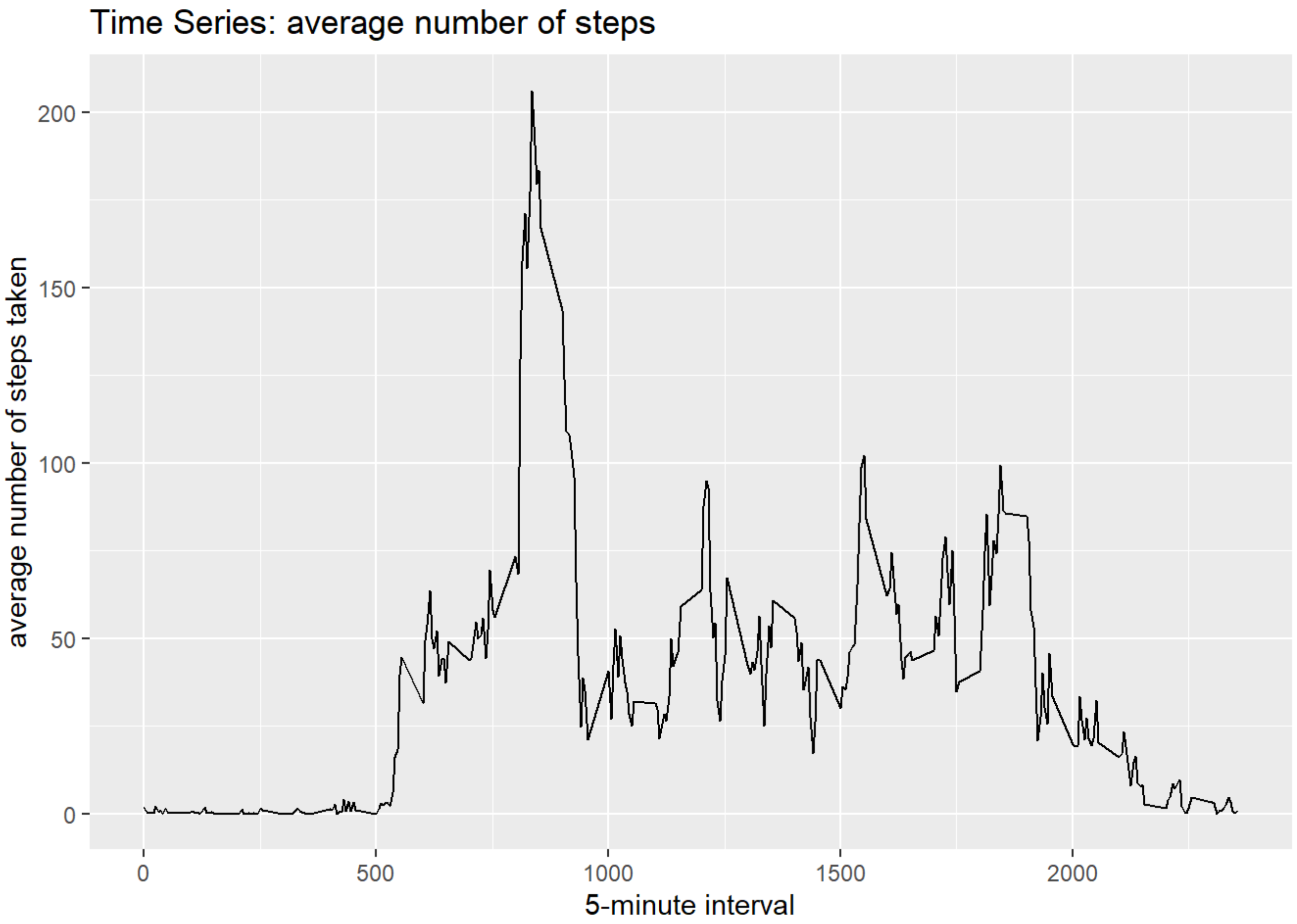
```
stepsMedian
```

```
## [1] 10395
```

```
library(ggplot2)

# Create the means by intervals
averages <- aggregate(x=list(steps=DF$steps), by=list(interval=DF$interval),FUN=mean, na.rm=T
RUE)

ggplot(data=averages, aes(x=interval, y=steps)) +
  geom_line() +
  ggtitle("Time Series: average number of steps") +
  xlab("5-minute interval") +
  ylab("average number of steps taken")
```



```
averages[which.max(averages$steps),]
```

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

```
# copy of data frame
DF2 <- DF

# add column for copleating index
DF2$CI <- "original"

# number of rows to check
l <- nrow(DF2)

# numbers of NAs
length(which(is.na(DF2$steps)))
```

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

```
# replace NAs by corresponing mean of the same interval --> complete data frame DF2
for (i in 1:l) {
  if (is.na(DF2[i,1])) {
    DF2[i,1] <- averages[averages$interval == DF2[i,3],2]
    DF2[i,4] <- "completed"
  }
}

# numbers of NAs / completed (control)
length(which(is.na(DF2$steps)))
```

```
## [1] 0
```

```
length(which(DF2$CI=="completed"))
```

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

```
# Recreate the sums of steps per date
DFsteps2 <- tapply(DF2$steps, DF2$date, FUN=sum, na.rm=TRUE )

# Recreate the mean and median of steps per date
stepsMean2 <- mean(DFsteps2)
stepsMedian2 <- median(DFsteps2)

c(stepsMean2, stepsMean)
```

```
## [1] 10766.19  9354.23
```

```
c(stepsMedian2, stepsMedian)
```

```
## [1] 10766.19 10395.00
```

```
# Preparation environment
library(ggplot2)
library(gridExtra)
```

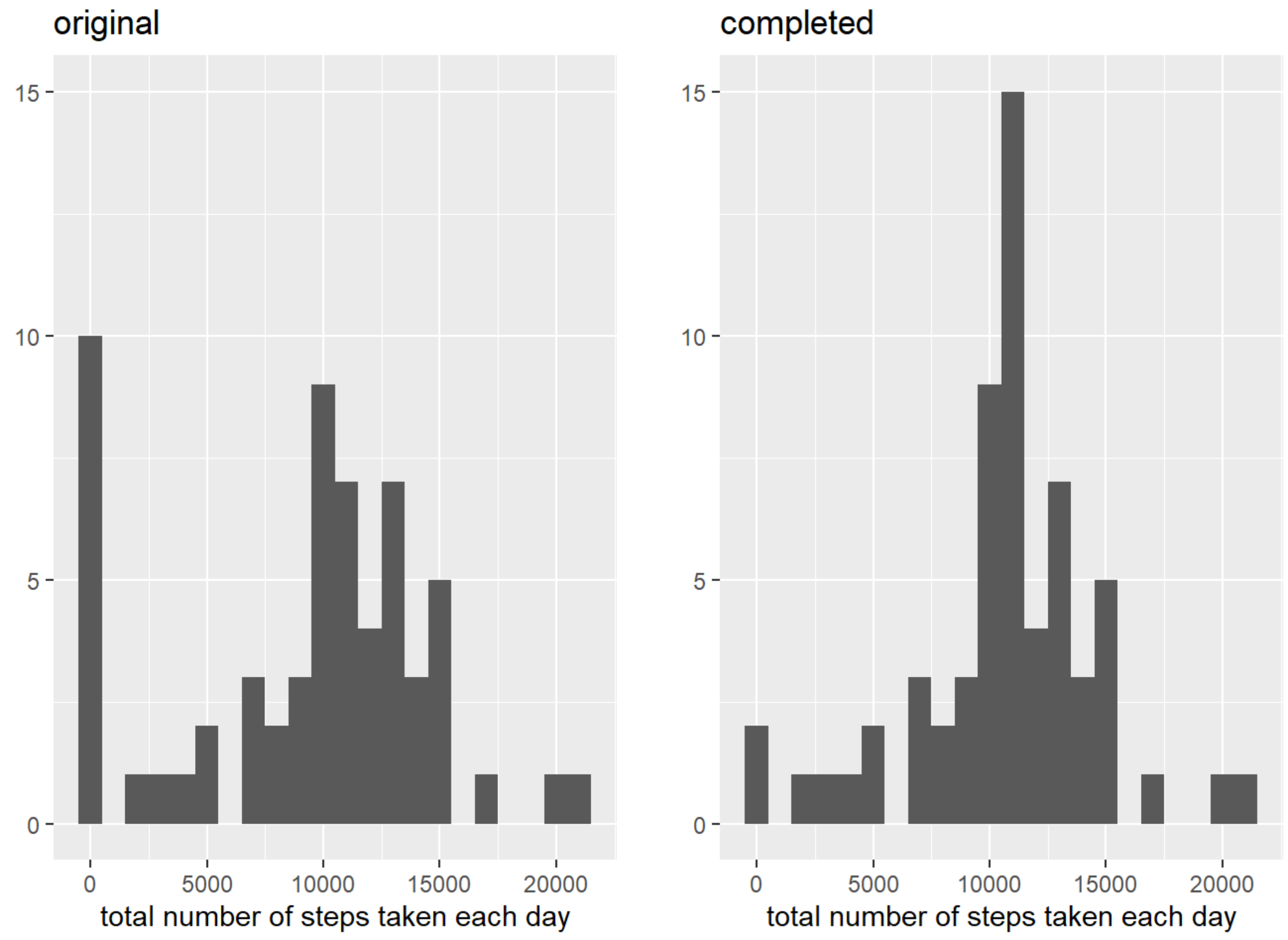
```
## Warning: package 'gridExtra' was built under R version 3.6.3
```

```
require(gridExtra)

# Perform histogram of steps per day
plot1 <- qplot(DFsteps,
               binwidth=1000,
               ylim=c(0,15),
               main="original",
               xlab="total number of steps taken each day")

plot2 <- qplot(DFsteps2,
               binwidth=1000,
               ylim=c(0,15),
               main="completed",
               xlab="total number of steps taken each day")

# Plotting 2 plot in grid
grid.arrange(plot1, plot2, ncol=2)
```



```
library(ggplot2)
library(gridExtra)

# Formatting and expanding DF2 by $WD (Weekday in German) an $WDG (WeekDayGroup)
DF2[,2] <- as.Date(DF2[,2])
DF2$WD <- weekdays(DF2[,2])
DF2$WDG <- "week"               # default = "week"

# Filling in the WeekDayGroup in German
for (i in 1:l) {
  if (DF2[i,5] == "Samstag" | DF2[i,5] == "Sonntag") {
    DF2[i,6] <- "weekend"
  }
}

DF2[,6] <- as.factor(DF2[,6])

DF2w <-subset(DF2,DF2[,6]=="week")
DF2we <-subset(DF2,DF2[,6]=="weekend")
```