R Data Programming Project

Thejas Raju

May 26, 2019

ANALYSIS

TASKS TO BE COMPLETED

1.1. Load the lawyers' data into R. What proportion of the lawyers practices litigation law? (Give your answer to 2 decimal places.)

1.2. Is the proportion of lawyers in the Boston office that practice corporate law higher

than the proportion of lawyers in the Providence office that practice corporate law?

1.3. Use the aggregate function to compute the average age of lawyers who practice corporate law and of lawyers who practice litigation law, across the different levels of seniority. Label the columns of the resulting data frame appropriately.

1.4. Which office has the youngest median age?

Task-1.1

```r
Lawyerdf <- read.table("Lawyers.csv",
                header = TRUE,
                sep = ",")

prop1<-prop.table(table(Lawyerdf$Practice))
round(prop1, 2)
```

```
##
##  Corporate Litigation
##       0.42       0.58
```

```r
sprintf("The proportion of Lawyers practicing Litigation law is:%0.02f",round(prop1[2],2))
```

```
## [1] "The proportion of Lawyers practicing Litigation law is:0.58"
```

Task-1.2

```r
prop2<-prop.table(table(Lawyerdf$Practice,Lawyerdf$Office))
prop2
```

```
##
##                   Boston    Harvard Providence
##   Corporate  0.26760563 0.11267606 0.04225352
##   Litigation 0.40845070 0.15492958 0.01408451
```

```r
prop2<-as.data.frame.matrix(prop2)
bostonCorp<-prop2['Corporate','Boston']
provCorp<-prop2['Corporate','Providence']
if(bostonCorp<provCorp)
{
  sprintf("Proportion of lawyers in the Providence offce that practice corporate law is greater than tha
} else
{
  sprintf("Proportion of lawyers in the Boston office that practice corporate law is greater than that
}
```

```
## [1] "Proportion of lawyers in the Boston office that practice corporate law is greater than that of
```

Task-1.3

```
aggdata1 <-data.frame(aggregate(Lawyerdf$Age, by=list(Lawyerdf$Seniority,Lawyerdf$Practice),
                    FUN=mean, na.rm=TRUE))
colnames(aggdata1)=c("Seniority Level","Branch of Law","Average Age")
aggdata1
```

```
##   Seniority Level Branch of Law Average Age
## 1       Associate     Corporate    36.71429
## 2         Partner     Corporate    48.50000
## 3       Associate    Litigation    34.61905
## 4         Partner    Litigation    47.70000
```

Task-1.4

```
aggdata2 <-data.frame(aggregate(Lawyerdf$Age, by=list(Lawyerdf$Office),
                    FUN=median, na.rm=TRUE))
colnames(aggdata2)=c("Office","MedianAge")
aggdata2
```

```
##       Office MedianAge
## 1     Boston      39.5
## 2    Harvard      38.0
## 3 Providence      46.0
```

```
lowestMedAge<-aggdata2$Office[aggdata2$MedianAge == min(aggdata2$MedianAge)]

sprintf('Office with the lowest median age is %s',lowestMedAge)
```

```
## [1] "Office with the lowest median age is Harvard"
```

WRITING OWN FUNCTION

TASKS TO BE COMPLETED

2.1. Write a function which compute the Rosenbrock banana function using a loop. Test the function on the vectors x = (.2, .5) and x = (.2, .5, .1, .6)

2.2. Propose an alternative function that does not use any loop. Test the function on the same two vectors.

2.3. Compare the timings you obtain by repeating the function calls 100 times using the vector x = (.2, .5, .1, .6) as input.

Task-2.1

```
x = c(0.2,0.5,0.1,0.6)
```

```
i=1
ans=0
objective <- function(x) {
  repeat
  {
    if(i==length(x))
    {
      return (ans)
      break
    }
  else
    {
      ans=ans+ (100 * (x[i+1] - x[i] * x[i])^2 + (1 - x[i])^2 )
      i=i+1
    }
  }
}
x<-c(0.2,0.5)
sprintf("The rosenbrock value for x<-c(0.2,0.5) is %0.1f",objective(x))
```

```
## [1] "The rosenbrock value for x<-c(0.2,0.5) is 21.8"
```

```
x<-c(0.2,0.5,0.1,0.6)
sprintf("The rosenbrock value for x<-c(0.2,0.5,0.1,0.6) is %0.1f",objective(x))
```

```
## [1] "The rosenbrock value for x<-c(0.2,0.5,0.1,0.6) is 59.9"
```

Task-2.2

```
rosen<- function(x,y)
{
  return(100 * (y - x * x)^2 + (1 - x)^2)
}
answer<-function(x)
{
  return(sum(mapply(rosen,c(x[1:(length(x)-1)]),c(x[c(2:(length(x)))]))))
}
x<-c(0.2,0.5)
sprintf("The rosenbrock value for x<-c(0.2,0.5) is %0.1f",answer(x))
```

```
## [1] "The rosenbrock value for x<-c(0.2,0.5) is 21.8"
```

```
x<-c(0.2,0.5,0.1,0.6)
sprintf("The rosenbrock value for x<-c(0.2,0.5,0.1,0.6) is %0.1f",answer(x))
```

```
## [1] "The rosenbrock value for x<-c(0.2,0.5,0.1,0.6) is 59.9"
```

Task-2.3

```
x<-c(0.2,0.5,0.1,0.6)
sprintf('Time taken by using function with loop')
```

```
## [1] "Time taken by using function with loop"
```

```r
system.time(for (k in 1:100) objective(x))
```

```
##    user  system elapsed
##    0.02    0.00    0.02
```

```r
sprintf('Time taken by using function without loop')
```

```
## [1] "Time taken by using function without loop"
```

```r
system.time(for (k in 1:100) answer(x))
```

```
##    user  system elapsed
##    0.02    0.00    0.01
```

WRITING S3 METHODS

TASKS TO BE COMPLETED

3.1. Load in the data as an object called DublinAirport. Assign to the DublinAiport object the classes WeatherData and data.frame.

3.2. Write an S3 summary method for an object of class WeatherData which produces the following statistical summaries for the rain, maxtp, mintp variables: mean, standard deviation, minimum, maximum.

3.3. Download the new data set 2018 09 Cork Airport.csv from Blackboard, assign the classes WeatherData and data.frame to the object containing the Cork data, and test your function on it. Interpret your findings for Dublin and Cork Airports.

3.4. Create an S3 plot method for the class WeatherData that produces the following plots.

. Two plots must be on a single panel, one above the other. Only the plot on the top panel will contain a main title.

. The plot on the top is about the daily Air Temperature (C). It must include the following:

- lines plot to show the daily air temperatures

- by default the plot will draw a red line for the maximum temperatures and a blue line for the minumum temperatures. The user must be able to change these colors

- the plot must include meaningful labels for the axis and legend

- the plot must include a grey vertical dotted line for each day to clearly identify the day corresponding to each couple of points.

- the plot by default should allow the user to identify clearly the noteworthy points by adding a point character over the value of the highest maximum temperature registered and a point character over the value of the lowest minimum temperature registered. The user must be able to decide to avoid to add the point characters to the plot.

. The plot on the bottom is about the daily Precipitation Amount (mm). It must include the following:

- vertical-line plot to show the daily precipitation amount

- by default the plot will draw the vertical bar for the day with the highest amount of rain in red. The user must be able to change the color to be used.

. Test your function on the Dublin and Cork airport data set, and set different meaningful titles for the two cases.

Task-3.1

```r
DublinAirport <- read.csv("2018_09_Dublin_Airport.csv",
                          , header = TRUE, fileEncoding="UTF-8-BOM")
sprintf("Loading of DublinAirport Dataset Complete")
```

```
## [1] "Loading of DublinAirport Dataset Complete"
```

```r
DublinAirport
```

```
##         date rain maxtp mintp
## 1  01-Sep-18  0.1  20.9  11.8
## 2  02-Sep-18  1.9  23.0  10.4
## 3  03-Sep-18  0.2  16.9   6.6
## 4  04-Sep-18  0.0  17.5   4.6
## 5  05-Sep-18  4.0  19.2   4.9
## 6  06-Sep-18  4.1  15.3   5.8
## 7  07-Sep-18  0.0  15.6   5.4
## 8  08-Sep-18  6.2  17.8  13.0
## 9  09-Sep-18  0.0  17.7  11.9
## 10 10-Sep-18  0.5  16.6  11.5
## 11 11-Sep-18  2.1  16.5  10.6
## 12 12-Sep-18  0.1  16.9   8.4
## 13 13-Sep-18  0.0  16.7   9.2
## 14 14-Sep-18  1.7  16.6  11.1
## 15 15-Sep-18  0.9  16.9  10.9
## 16 16-Sep-18  2.3  18.7  10.1
## 17 17-Sep-18  0.1  20.4  10.0
## 18 18-Sep-18  1.5  21.1  13.6
## 19 19-Sep-18  0.9  17.9  10.3
## 20 20-Sep-18 15.7  12.0   6.0
## 21 21-Sep-18  1.0  14.1   6.1
## 22 22-Sep-18  0.0  11.9   3.4
## 23 23-Sep-18  0.0  13.3   3.9
## 24 24-Sep-18  0.0  13.9   3.5
## 25 25-Sep-18  0.0  15.9   0.4
## 26 26-Sep-18  0.0  19.5  13.4
## 27 27-Sep-18  0.5  16.6   4.2
## 28 28-Sep-18  0.0  13.7   1.5
## 29 29-Sep-18  0.0  14.8   1.9
## 30 30-Sep-18  0.0  12.8   5.1
```

```r
sprintf("Assigning Classes WeatherData and data.frame ")
```

```
## [1] "Assigning Classes WeatherData and data.frame "
```

```r
class(DublinAirport) <- 'WeatherData'
DublinAirport
```

```
## $date
##  [1] 01-Sep-18 02-Sep-18 03-Sep-18 04-Sep-18 05-Sep-18 06-Sep-18 07-Sep-18
##  [8] 08-Sep-18 09-Sep-18 10-Sep-18 11-Sep-18 12-Sep-18 13-Sep-18 14-Sep-18
## [15] 15-Sep-18 16-Sep-18 17-Sep-18 18-Sep-18 19-Sep-18 20-Sep-18 21-Sep-18
## [22] 22-Sep-18 23-Sep-18 24-Sep-18 25-Sep-18 26-Sep-18 27-Sep-18 28-Sep-18
## [29] 29-Sep-18 30-Sep-18
## 30 Levels: 01-Sep-18 02-Sep-18 03-Sep-18 04-Sep-18 05-Sep-18 ... 30-Sep-18
##
## $rain
##  [1]  0.1  1.9  0.2  0.0  4.0  4.1  0.0  6.2  0.0  0.5  2.1  0.1  0.0  1.7
## [15]  0.9  2.3  0.1  1.5  0.9 15.7  1.0  0.0  0.0  0.0  0.0  0.0  0.5  0.0
## [29]  0.0  0.0
##
## $maxtp
##  [1] 20.9 23.0 16.9 17.5 19.2 15.3 15.6 17.8 17.7 16.6 16.5 16.9 16.7 16.6
## [15] 16.9 18.7 20.4 21.1 17.9 12.0 14.1 11.9 13.3 13.9 15.9 19.5 16.6 13.7
## [29] 14.8 12.8
##
## $mintp
##  [1] 11.8 10.4  6.6  4.6  4.9  5.8  5.4 13.0 11.9 11.5 10.6  8.4  9.2 11.1
## [15] 10.9 10.1 10.0 13.6 10.3  6.0  6.1  3.4  3.9  3.5  0.4 13.4  4.2  1.5
## [29]  1.9  5.1
##
## attr(,"class")
## [1] "WeatherData"
## attr(,"row.names")
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
## [24] 24 25 26 27 28 29 30
```

```
class(DublinAirport) <- 'data.frame'
DublinAirport
```

```
##         date rain maxtp mintp
## 1  01-Sep-18  0.1  20.9  11.8
## 2  02-Sep-18  1.9  23.0  10.4
## 3  03-Sep-18  0.2  16.9   6.6
## 4  04-Sep-18  0.0  17.5   4.6
## 5  05-Sep-18  4.0  19.2   4.9
## 6  06-Sep-18  4.1  15.3   5.8
## 7  07-Sep-18  0.0  15.6   5.4
## 8  08-Sep-18  6.2  17.8  13.0
## 9  09-Sep-18  0.0  17.7  11.9
## 10 10-Sep-18  0.5  16.6  11.5
## 11 11-Sep-18  2.1  16.5  10.6
## 12 12-Sep-18  0.1  16.9   8.4
## 13 13-Sep-18  0.0  16.7   9.2
## 14 14-Sep-18  1.7  16.6  11.1
## 15 15-Sep-18  0.9  16.9  10.9
## 16 16-Sep-18  2.3  18.7  10.1
## 17 17-Sep-18  0.1  20.4  10.0
## 18 18-Sep-18  1.5  21.1  13.6
## 19 19-Sep-18  0.9  17.9  10.3
## 20 20-Sep-18 15.7  12.0   6.0
## 21 21-Sep-18  1.0  14.1   6.1
## 22 22-Sep-18  0.0  11.9   3.4
## 23 23-Sep-18  0.0  13.3   3.9
## 24 24-Sep-18  0.0  13.9   3.5
## 25 25-Sep-18  0.0  15.9   0.4
## 26 26-Sep-18  0.0  19.5  13.4
## 27 27-Sep-18  0.5  16.6   4.2
## 28 28-Sep-18  0.0  13.7   1.5
## 29 29-Sep-18  0.0  14.8   1.9
## 30 30-Sep-18  0.0  12.8   5.1
```

Task-3.2

```
print.WeatherData<-function(weather)
{
  cat('Statistical Summary \n')
  cat('Mean:',mean(weather$rain),'\n')
  cat('Median:',median(weather$rain),'\n')
  cat('Min:',min(weather$rain),'\n')
  cat('Median:',max(weather$rain),'\n')
  cat('\nMintp Statistics \n')
  cat('Mean:',mean(weather$mintp ),'\n')
  cat('Median:',median(weather$mintp),'\n')
  cat('Min:',min(weather$mintp),'\n')
  cat('Median:',max(weather$mintp),'\n')


  cat('\nMaxtp Statistics \n')
  cat('Mean:',mean(weather$maxtp),'\n')
  cat('Median:',median(weather$maxtp),'\n')
  cat('Min:',min(weather$maxtp),'\n')
  cat('Median:',max(weather$maxtp),'\n')
}

print.WeatherData(DublinAirport)
```

```
## Statistical Summary
## Mean: 1.46
## Median: 0.15
## Min: 0
## Median: 15.7
##
## Mintp Statistics
## Mean: 7.65
## Median: 7.5
## Min: 0.4
## Median: 13.6
##
## Maxtp Statistics
## Mean: 16.69
## Median: 16.65
## Min: 11.9
## Median: 23
```

Task-3.3

```
CorkAirport <- read.csv("2018_09_Cork_Airport.csv",
                        , header = TRUE, fileEncoding="UTF-8-BOM")
sprintf("Loading of CorkAirport Dataset Complete")
```

```
## [1] "Loading of CorkAirport Dataset Complete"
```

```
CorkAirport
```

```
##         date rain maxtp mintp
## 1  01-Sep-18  0.4  20.2  13.0
## 2  02-Sep-18  4.8  19.7  12.0
## 3  03-Sep-18  0.0  16.8   8.8
## 4  04-Sep-18  0.0  16.3   6.5
## 5  05-Sep-18  0.0  17.4   8.9
## 6  06-Sep-18  0.6  16.3  11.9
## 7  07-Sep-18  1.1  17.6  10.5
## 8  08-Sep-18 11.1  15.7  13.0
## 9  09-Sep-18  1.0  18.2   9.2
## 10 10-Sep-18  1.3  16.3   8.2
## 11 11-Sep-18  4.5  16.7   9.9
## 12 12-Sep-18  0.1  17.0   8.1
## 13 13-Sep-18  0.0  15.2   8.0
## 14 14-Sep-18  0.1  17.6  10.0
## 15 15-Sep-18  2.5  16.2  10.5
## 16 16-Sep-18  7.2  17.7   9.4
## 17 17-Sep-18 12.5  17.1   9.3
## 18 18-Sep-18  1.0  17.8  11.5
## 19 19-Sep-18  2.3  14.7   9.4
## 20 20-Sep-18 19.2  10.3   6.2
## 21 21-Sep-18  0.2  15.1   4.9
## 22 22-Sep-18  6.8  10.5   7.3
## 23 23-Sep-18  0.2  13.2   6.3
## 24 24-Sep-18  0.0  13.2   4.9
## 25 25-Sep-18  0.0  14.4   5.8
## 26 26-Sep-18  0.0  18.5   9.2
## 27 27-Sep-18  0.1  17.9   9.4
## 28 28-Sep-18  0.0  13.9   6.3
## 29 29-Sep-18  0.0  14.0   4.7
## 30 30-Sep-18  0.4  13.0   7.5
```

```r
sprintf("Assigning Classes WeatherData and data.frame ")
```

```
## [1] "Assigning Classes WeatherData and data.frame "
```

```r
class(CorkAirport) <- 'WeatherData'
CorkAirport
```

```
## Statistical Summary
## Mean: 2.58
## Median: 0.4
## Min: 0
## Median: 19.2
##
## Mintp Statistics
## Mean: 8.686667
## Median: 9.05
## Min: 4.7
## Median: 13
##
## Maxtp Statistics
## Mean: 15.95
## Median: 16.3
## Min: 10.3
## Median: 20.2
```

```r
class(CorkAirport) <- 'data.frame'
CorkAirport
```

```
##         date rain maxtp mintp
## 1  01-Sep-18  0.4  20.2  13.0
## 2  02-Sep-18  4.8  19.7  12.0
## 3  03-Sep-18  0.0  16.8   8.8
## 4  04-Sep-18  0.0  16.3   6.5
## 5  05-Sep-18  0.0  17.4   8.9
## 6  06-Sep-18  0.6  16.3  11.9
## 7  07-Sep-18  1.1  17.6  10.5
## 8  08-Sep-18 11.1  15.7  13.0
## 9  09-Sep-18  1.0  18.2   9.2
## 10 10-Sep-18  1.3  16.3   8.2
## 11 11-Sep-18  4.5  16.7   9.9
## 12 12-Sep-18  0.1  17.0   8.1
## 13 13-Sep-18  0.0  15.2   8.0
## 14 14-Sep-18  0.1  17.6  10.0
## 15 15-Sep-18  2.5  16.2  10.5
## 16 16-Sep-18  7.2  17.7   9.4
## 17 17-Sep-18 12.5  17.1   9.3
## 18 18-Sep-18  1.0  17.8  11.5
## 19 19-Sep-18  2.3  14.7   9.4
## 20 20-Sep-18 19.2  10.3   6.2
## 21 21-Sep-18  0.2  15.1   4.9
## 22 22-Sep-18  6.8  10.5   7.3
## 23 23-Sep-18  0.2  13.2   6.3
## 24 24-Sep-18  0.0  13.2   4.9
## 25 25-Sep-18  0.0  14.4   5.8
## 26 26-Sep-18  0.0  18.5   9.2
## 27 27-Sep-18  0.1  17.9   9.4
## 28 28-Sep-18  0.0  13.9   6.3
## 29 29-Sep-18  0.0  14.0   4.7
## 30 30-Sep-18  0.4  13.0   7.5
```

```r
sprintf("It can be observed that the previously defined WeatherData function is used to produce the Sta
```

```
## [1] "It can be observed that the previously defined WeatherData function is used to produce the Stat
```

Task-3.4

Entering choice of city

```r
choice1<-'b'
choice1<-readline(prompt="Please enter whether you want to view Dublin stats OR Cork Stats Or Both (Ent
```

```
## Please enter whether you want to view Dublin stats OR Cork Stats Or Both (Enter D/C/B):
```

Checking choice of city

```r
if(choice1==('D')||choice1==('C')||choice1==('B')||choice1==('d')||choice1==('b')||choice1==('c'))
{
  sprintf('Correct Input. Option Chosen:%s',choice1)
}else
{
  sprintf('Incorrect Option Entered. Please Try Again')
}
```

```
## [1] "Incorrect Option Entered. Please Try Again"
```

Defining Colors to be used and function to check if string is a valid color

```
maxtpColor<-""
mintpColor<-""
isColor <- function(x)
{
  res <- try(col2rgb(x),silent=TRUE)
  return(!"try-error"%in%class(res))
}
```

Entering maxtp color

```
maxtpColor<- readline(prompt = "Enter Color for Maxtp:")
```

```
## Enter Color for Maxtp:
```

Entering mintp color

```
mintpColor<- readline(prompt = "Enter Color for Mintp:")
```

```
## Enter Color for Mintp:
```

Checking if valid colors are entered

```
if(isColor(maxtpColor) && isColor(mintpColor))
{
  writeLines("Correct color values")
}else
{
  writeLines("Incorrect color values. Default colors will be plotted")
  maxtpColor<-"red"
  mintpColor<-"blue"
}
```

```
## Incorrect color values. Default colors will be plotted
```

Function to plot charts

```
plotting<- function(mintpColor,maxtpColor,point,df, city, rainc)
{
x<-df$date; y1=df$maxtp; y2=df$mintp
par(mfcol = c(2, 1))
par(mar = c(3, 3, 2, 1), mgp = c(2, 0.7, 0), las = 1)
title<- paste("Air Temperature and Daily Precipitation in",city)
plot(y1,type="l", pch=ifelse(x==max.col(y1),2,0), col=maxtpColor, xlab="September 2018", ylab="Temperatu
lines(x, y2, pch=18, col=mintpColor, type="l", lty=1)
legend("topright", c("Maxtp","Mintp"),fill = c(maxtpColor,mintpColor), col=c(maxtpColor,mintpColor),tex
abline(v = x, col="grey", lwd=1, lty=2)
mf <- data.frame(df)
```

```
if(point==1)
{
mf2<-mf[ mf$maxtp == max(mf$maxtp), ]
mf3<-mf[ mf$mintp == min(mf$mintp), ]
points(mf2$date,mf2$maxtp, pch=8)
points(mf3$date,mf3$mintp, pch=8)
}
#Below works
mf1<-mf[mf$rain==max(mf$rain),]
plot(df$rain, type="h", col=ifelse(x==mf1$date, rainc , "black"), xlab = "September 2018", ylab = "Preci
}
```

Choice to view Point in chart

```
choice2<-readline(prompt="Please enter whether you want to view point characters on the plot. Enter Y fo
```

```
## Please enter whether you want to view point characters on the plot. Enter Y for Yes and N for No.
```

```
point=0
```

Checking whether valid choice is entered

```
if(choice2=="Y"||choice2=="y"||choice2=="N"||choice2=="n")
{
  point=1
  sprintf('Point will be plotted')

}else
{
  sprintf('Incorrect Option. Point will not be plotted')
}
```

```
## [1] "Incorrect Option. Point will not be plotted"
```

Choice to change color of max rain bar in chart

```
choice3<-readline(prompt="Please enter the color to be plotted to indicate maximum rain:")
```

```
## Please enter the color to be plotted to indicate maximum rain:
```

```
raincolor<-'red'
```

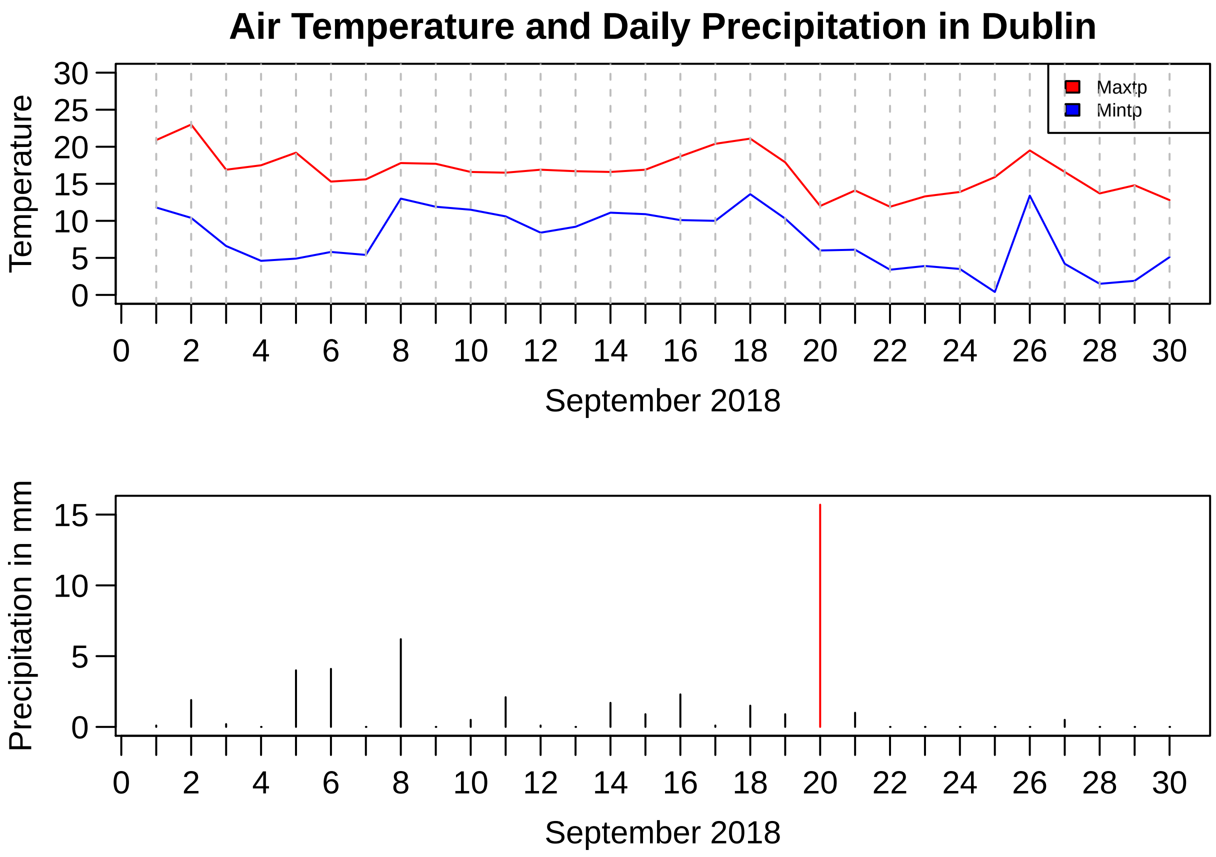
Checking whether valid choice is entered

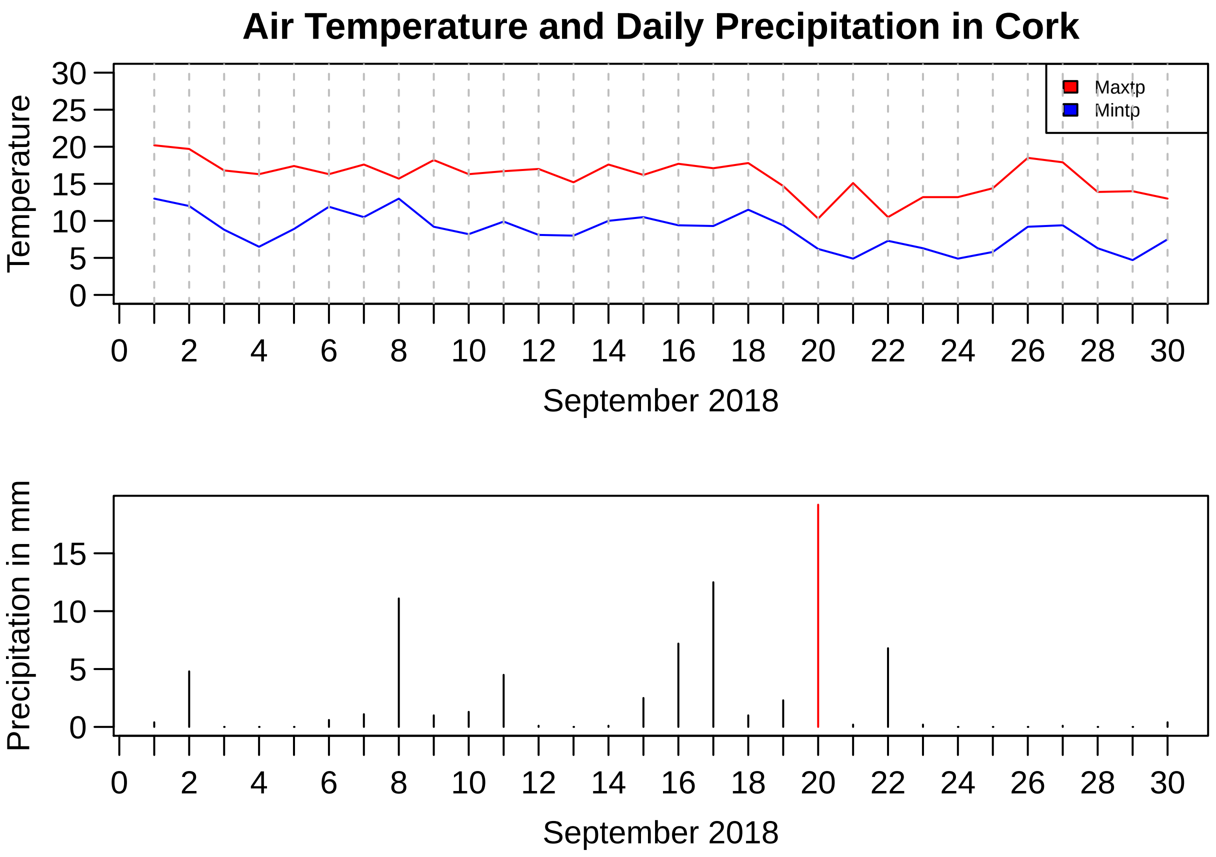
```
if(isColor(choice3))
{
  writeLines("Color of Maximum Rain bar will be changed")
  raincolor<-choice3
}else
{
  writeLines("Incorrect color values. Default color will be plotted")
}
```

```
## Incorrect color values. Default color will be plotted
```

Plotting the charts

```
if(choice1=='D'||choice1=='d')
{
  df<- read.csv("2018_09_Dublin_Airport.csv",
                , header = TRUE, fileEncoding="UTF-8-BOM")
  plotting(mintpColor, maxtpColor, point, df, "Dublin", raincolor)
} else if(choice1=='c'||choice1=='C')
{
  df<-read.csv("2018_09_Cork_Airport.csv",
               , header = TRUE, fileEncoding="UTF-8-BOM")
  plotting(mintpColor, maxtpColor, point, df, "Cork",raincolor)
}else
{
  df<- read.csv("2018_09_Dublin_Airport.csv",
                , header = TRUE, fileEncoding="UTF-8-BOM")
  plotting(mintpColor, maxtpColor, point, df, "Dublin",raincolor)
  df<-read.csv("2018_09_Cork_Airport.csv",
               , header = TRUE, fileEncoding="UTF-8-BOM")
  plotting(mintpColor, maxtpColor, point, df, "Cork",raincolor)
}
```


Air Temperature and Daily Precipitation in Dublin
Temperature
30
25
20
15
10
5
0
Maxtp
Mintp
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
September 2018
Precipitation in mm
15
10
5
0
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
September 2018


Air Temperature and Daily Precipitation in Cork
Temperature
0
5
10
15
20
25
30
🟥 Maxtp
🟦 Mintp
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
September 2018
Precipitation in mm
0
5
10
15
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
September 2018