MATH215 Assignment 1

Pierre Visconti

Problem 1:

```
help("AirPassengers")
```

```
## starting httpd help server ... done
```

```
class(AirPassengers)
```

```
## [1] "ts"
```

```
str(AirPassengers)
```

```
##  Time-Series [1:144] from 1949 to 1961: 112 118 132 129 121 135 148 148 136 119 ...
```

```
summary(AirPassengers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   104.0   180.0   265.5   280.3   360.5   622.0
```

The values in this dataset represent monthly total of international airline passengers in thousands from 1949 to 1960. The class command returns the value ts which means that the dataset is a time series. The str command also confirms it as a time series with a range of 1:144, meaning the dataset has 144 rows, or in other words: has a length of 144.This command also prints out some of the first rows of the dataset which shows us that there is a single number for each row, meaning we can treat this dataset as a vector since it has only a single column.
The summary command gives us the statistics of a box plot for the dataset. With this command we now know the median, the range where most of the data falls into (1st quartile to 3rd quartile), and the maximum values (which in this case shows us that there are some extreme outliers).

```
in.1950 = AirPassengers[13:24]
print(in.1950)
```

```
##  [1] 115 126 141 135 125 149 170 170 158 133 114 140
```

```
total.1950 = sum(in.1950)
print(total.1950)
```

```
## [1] 1676
```

```
which.max(in.1950)
```

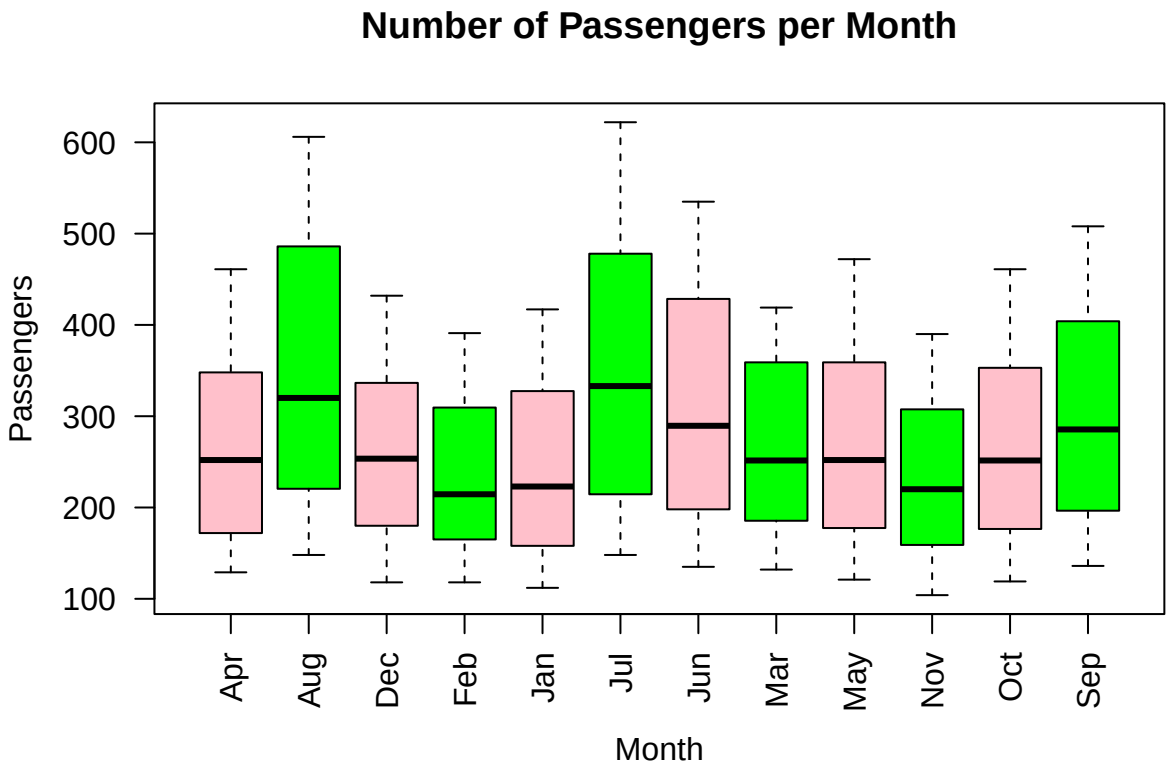
```
## [1] 7
```

```
which.max(in.1950%%1000000)
```

```
## [1] 7
```

Problem 2:

```
mon = c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")
Mon = rep(mon, 12)
pass = as.vector(AirPassengers)
air.data = data.frame(pass, Mon)
colnames(air.data) = c("Passengers", "Month")
boxplot(Passengers~Month, data=air.data, main="Number of Passengers per Month", col= c("pink", "green")
```



The practice of hiring extra personnel and ordering additional supplies to accommodate a higher number of passengers for the month of December was not justified as there was not a particularly larger number of passengers travelling that month compared to the other months of the year. In fact, December had much lower rates of travel compared to months like August and July.

Problem 3:

```
marathon = read.csv("marathon_results_2015.csv")
mean(marathon$Official.Time)
```

```
## Warning in mean.default(marathon$Official.Time): argument is not numeric or
## logical: returning NA
```

```
## [1] NA
```

```
class(marathon$Official.Time)
```

```
## [1] "character"
```

```
print(marathon$Official.Time[1:10])
```

```
##  [1] "2:09:17" "2:09:48" "2:10:22" "2:10:47" "2:10:49" "2:10:52" "2:11:20"
##  [8] "2:12:42" "2:13:35" "2:13:52"
```

The mean command does not work because Official.Time contains characters and not numbers. By printing the first 10 indexes we can see that each entry is contained within quotation marks, indicating they are strings and not numbers.

```
temp = marathon$Official.Time[1]
temp2 = strptime(temp, "%H:%M:%S")
temp3 = format(strptime(temp, "%H:%M:%S"), "%H")
tempH <- as.numeric(format(strptime(temp,"%H:%M:%S"),"%H"))
print(temp)
```

```
## [1] "2:09:17"
```

```
print(temp2)
```

```
## [1] "2023-01-17 02:09:17 PST"
```

```
print(temp3)
```

```
## [1] "02"
```

```
print(tempH)
```

```
## [1] 2
```

temp, temp2, and temp3 are all character objects. tempH is a numerical object. The first command for temp2 added a date to the original value. The second command only assigned what was in the "H" section of the character to temp3, as was defined by "%H:%M:%S". The last command does the same thing as the second but then converts it into a numerical object and assigns it to tempH.

```
tempM = as.numeric(format(strptime(temp,"%H:%M:%S"),"%M"))
tempS = as.numeric(format(strptime(temp,"%H:%M:%S"),"%S"))
tempFinal = tempH + tempM/60 + tempS/60/60
print(tempFinal)
```

```
## [1] 2.154722
```

```
convert = function(x) {
  x.H = as.numeric(format(strptime(x,"%H:%M:%S"),"%H"))
  x.M = as.numeric(format(strptime(x,"%H:%M:%S"),"%M"))
  x.S = as.numeric(format(strptime(x,"%H:%M:%S"),"%S"))
  return(x.H + x.M/60 + x.S/60/60)

}
marathon$New.Time = sapply(marathon$Official.Time,FUN=convert, USE.NAMES=FALSE)
mean(marathon$New.Time, na.rm = T)
```

```
## [1] 3.773805
```

Problem 4:

```
unique(marathon$Country)
```

```
##  [1] "ETH" "KEN" "USA" "UKR" "RSA" "ITA" "RUS" "JPN" "CAN" "BEL" "NZL" "BLR"
## [13] "AUS" "GBR" "CRO" "ECU" "GER" "ESP" "SWE" "BRA" "HKG" "MEX" "DEN" "MAS"
## [25] "IRL" "ISL" "CHI" "GUA" "FIN" "SVK" "COL" "SUI" "CHN" "AUT" "NED" "FRA"
## [37] "CRC" "CYP" "POL" "NOR" "KOR" "POR" "TPE" "PER" "SIN" "PAN" "VEN" "LUX"
## [49] "CZE" "VIE" "MAR" "BER" "ROU" "IND" "LIE" "ARG" "ESA" "DOM" "ISR" "GRE"
## [61] "SLO" "LTU" "URU" "CAY" "EST" "JAM" "UAE" "VGB" "TUR" "LAT" "AND" "OMA"
## [73] "BAH" "TRI" "INA" "AHO" "PHI" "UGA" "QAT"
```

```
length(unique(marathon$Country))
```

```
## [1] 79
```

```
runners = rep(0, nrow(marathon))
(length(runners[which(marathon$Country=="USA")]) / length(marathon$Country)) * 100
```

```
## [1] 82.26182
```

There are 79 countries present in the data and 82.26% of runners are domestic.

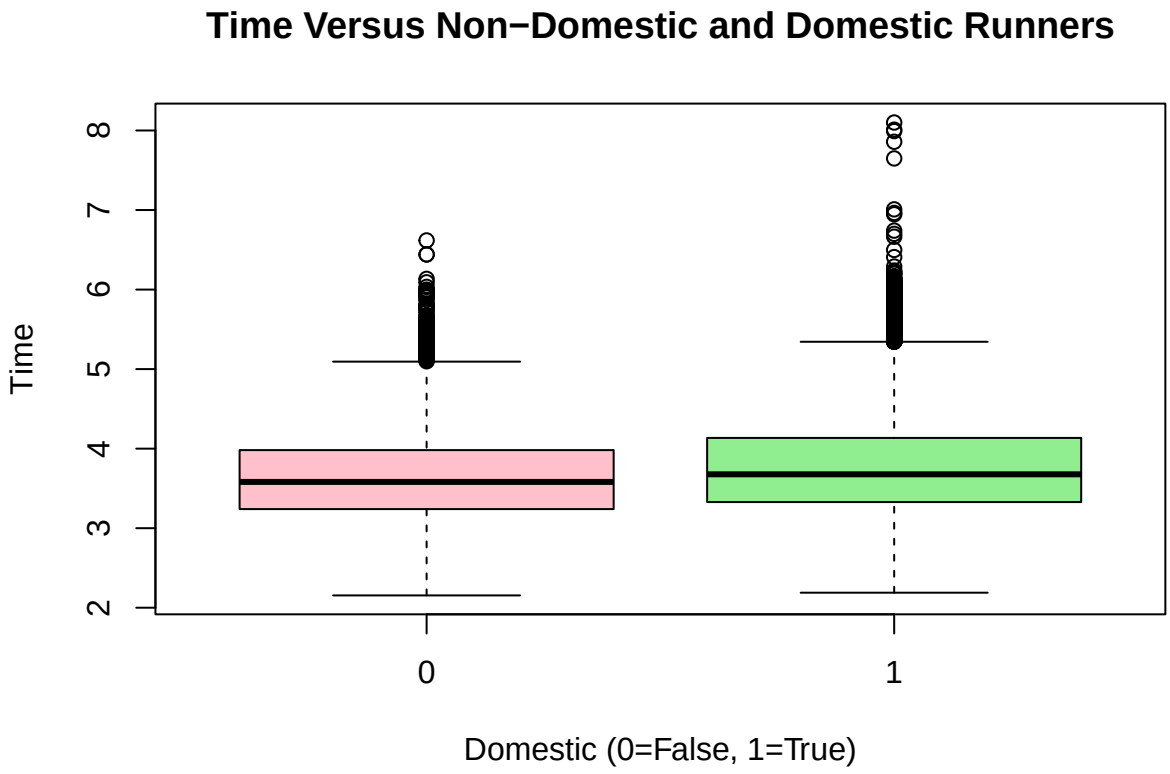
```
runners[which(marathon$Country=="USA")] = 1
marathon.ID = data.frame(domestic=runners, time=marathon$New.Time)

summary(marathon.ID$time[marathon.ID$domestic==0])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.155   3.240   3.582   3.672   3.982   6.618
```

```
summary(marathon.ID$time[marathon.ID$domestic==1])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   2.189   3.328   3.677   3.796   4.135   8.100       1
```

```
boxplot(time ~ domestic, data=marathon.ID, na.rm=T, main="Time Versus Non-Domestic and Domestic Runners
```



The belief that non-domestic runners are faster than domestic runners is indeed true, as supported by the data set. The median times for both groups is actually very close as well as where the majority of runners fall into for both groups. The domestic group had outliers that were much slower than the outliers from the non-domestic group which could help support the theory that only more professional/serious runners make up the non-domestic group.